DOCUMENT RESUME

ED 223 239 IR 010 493

TITLE	AEDS Proceedings: The Tomorrow in New Technology; Frontiers in Administrative Computing; Adventures in Instructional Computing.
INSTITUTION	Association for Educational Data Systems, Washington, D.C.
PUB DATE	82
NOTE	513p.; For related document, see ED 201 410.
AVAILABLE FROM	Association for Educational Data Systems, 1201 16th St., N.W., Washington, DC 20036 ($15.00).
PUB TYPE	Collected Works - Conference Proceedings (021) -- Viewpoints (120) -- Reports - General (140)
EDRS PRICE	MF02 Plus Postage. PC Not Available from EDRS.
DESCRIPTORS	*Computer Assisted Instruction; Computer Assisted Testing; Computer Graphics; *Computer Literacy; *Computer Managed Instruction; *Computer Oriented Programs; *Computer Programs; Educational Technology; Elementary Secondary Education; Higher Education; Input Output Devices; *Microcomputers; Online Systems; Programing Languages; Simulation; Special Education

ABSTRACT

The 122 papers in this collection were presented in 15 sessions of the 20th annual convention of the Association for Educational Data Systems which was held in Orlando, Florida, May 10-14, 1982. Individual papers covered a wide variety of topics, including computer assisted instruction, computer managed instruction, computer literacy, instructional and administrative uses of microcomputers, software evaluation, computer assisted testing, use of computers with various handicaps and in specific subject areas, teacher education, information dissemination and library automation, computer graphics, and teaching computer programming. The AEDS board of directors, 1982 convention team, affiliate presidents and representatives, institutional members, and sustaining members are listed in the front of the book. (LMM)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED223239

Association For Educational Data Systems

AEDS

PROCEEDINGS

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION
CENTER (ERIC)

- This document has been reproduced as received from the person or organization originating it.
Minor changes have been made to improve reproduction quality.
- Points of view or opinions stated in this document do not necessarily represent official NIE position or policy

FRONTIERS IN ADMINISTRATIVE COMPUTING

ADVENTURES IN INSTRUCTIONAL COMPUTING

1 9 8 2

THE TOMORROW IN NEW TECHNOLOGY

Library of Congress Catalog Card Number ISSN 0147-9296

Copywright (C) 1982, Association for Educational Data Systems

IR010493

The Cover --
The Sheraton Twin Towers is the AEDS-'82 Convention Site. Additional areas of interest and convention activities are Sea World, Disney World, Circus World and The Space Center.

"PERMISSION TO REPRODUCE THIS MATERIAL IN MICROFICHE ONLY HAS BEEN GRANTED BY
Shirley Easterwood

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

FORWARD

The Association for Educational Data Systems (AEDS) is a private, non-profit international educational organization founded in 1962 by a group of professional educators and technical specialists in educational application of computers and technology. The aim of the organization is to provide a forum for the exchange of ideas and information about the relationship of modern technology to education.

AEDS membership is open to all persons interested in learning about educational data processing and who want to keep abreast of current developments and directions in educational data systems and computer technology. Included in the membership are educators and technical specialists from all over the United States, Canada, and other countries representing public and private schools, higher education, federal government, the military, and other professional and technical groups interested in the educational use of computers and technology.

AEDS-'82, the Association's 20th Annual Convention, is designed to bring together professionals who are interested in learning and sharing information about the latest developments in technology as it applies to the educational process. These proceedings represent a significant contribution to the growing body of knowledge in this vital area. The AEDS-'82 Convention Team is grateful to the many contributors to these proceedings. The Convention Team firmly believes this collection of papers can have considerable impact upon the educational community if all participants give each paper the careful attention it is due. Additional copies of these proceedings can be obtained for $15.00 from:

The Association for Educational Data Systems
1201 Sixteenth Street N.W.
Washington, D.C. 20036

The Convention Committee would like to express its appreciation for the support and assistance of all of those who have shown an interest in this year's Convention. The AEDS Board of Directors, Florida Association of Educational Data Systems (FAEDS), the membership, private industry and other interested persons have exhibited interest and initiative which are essential to making the Convention a success.

Most especially, I would like to thank all of the members of the AEDS-'82 Convention Team who have worked so tirelessly on the many details associated with this Convention.

Randall L. Johns
Convention Coordinator

ACKNOWLEDGEMENTS

AEDS BOARD OF DIRECTORS
1981-82

PRESIDENT

Dr. Dennis Spuck
Prof/Chrm. Dept. of Adm.
and Supv.
AED/FH University of Houston
Houston, TX 77004
713/749-7281

PRESIDENT-ELECT

Mr. A. V. Majors
Fort Worth Public Schools
3210 West Lancaster
Fort Worth, TX 76107
817/336-8691

PAST-PRESIDENT

Dr. Winston C. Addis
Supt. of Schools
Mt. Vernon Community Schools
Mt. Vernon, IA 52314
319/895-8845

EXECUTIVE SECRETARY

Ms. Shirley Easterwood
AEDS
1201 16th St., N. W.
Washington, D. C. 20036
202/822-7845

SECRETARY

Ms. Carolyn Meinhardt (1982)
San Diego City Sch. Educ. Ctr.
4100 Normal St., Room 1017
San Diego, CA 92103
714/293-8379

TREASURER

Mr. Steve Raucher (1982)
Montgomery Co. Pub. Schs.
850 Hungerford Drive
Rockville, MD 20850
301/279-3581

Dr. William C. Bozeman (1984)
210 Lindquist Ctr. Measurement
University of Iowa
Iowa City, IA 52242
319/353-5718

Mr. John W. Kelly (1984)
Palm Beach Junior College
4200 Congress Avenue
Lake Worth, FL 33460
305/439-8040

Dr. Kenneth Hasledalen (ex-officio)
Geographic Affiliate Rep.
MN SD Data Proc. Joint Board
1925 W. County RD B2
St. Paul, MN 55113
612/633-9110

Dr. D. Thomas King (1982)
St. Paul Public Schools
360 Colborne Street
St. Paul, MN 55102
612/293-7709

Mr. Donald Costello (1984)
Univ. of NE
Regents Hall
3835 Holdrege Street
Lincoln, NE 68583
402/472-3765

Mr. Rick Meyer (1983)
Mgr. Educ. Computer Services
Maricopa Community Colleges
3910 E. Washington St.
Phoenix, AZ 85034
602/244-8355 X340

Mr. Arthur Daniels, Jr. (1983)
COMPUWARE
29433 Southfield Road
Southfield MI 48076
313/559-0890 X566

Dr. John Hamblen (ex-officio)
AFIPS Representative
National Bureau of Standards
Center for Applied Math
Tech. Bldg. Room A151
Washington, D. C. 20234
301/921-2905

Mr. Alan Olkes (ex-officio)
Functional Affiliate Rep.
Dade Co. Public Schools
13135 Coral Way
Miami, FL 33175
305/552-9894

Mr. Howard Schumacher (1982)
Dir. Mgmt. Information
Proviso Twp. High School
807 S. 1st Avenue
Maywood, IL 60153
312/344-7000 X317

AEDS - 82 CONVENTION TEAM

Randall L. Johns
Convention Coordinator
Dir. of Data Processing
The School Board of Seminole County
1211 S. Mellonville Ave.
Sanford, FL 32771
305/322-1252

William E. McTammany
Asst. Convention Coordinator
Dir. of Data Processing
Fla. Jr. College
210 N. Maine St.
Jacksonville, FL 32202
904/358-1812

John G. Bolin
Activities Chairperson
Associate Superintendent of
Management & Personnel Services
Orange County Public Schools
P.O. Box 271
Orlando, FL 32802
305/423-9243

David Brittain
Program Chairperson
Dir. of the Fla. Educational
Computing Project
Department of Education
Knott Building
Tallahassee, FL 32301
305/488-0980

Terry Shoultes
Publicity Chairperson
Database Administrator
Palm Beach Comm. Public Schools
3323 Belvedere Rd.
West Palm Beach, FL 33402
305/683-0050

Bruce Collins
Exh. & Facil. Chairperson
Asst. Dir. of Data Management
Broward County Public Schools
P.O. Box 5408
Ft. Lauderdale, FL 33310
305/765-6370

Bernard Slessinger
Finance Chairperson
Registrar's Office
University of Central Florida
P.O. Box 25000
Orlando, FL 32816
305/275-2531

AFFILIATE PRESIDENTS AND REPRESENTATIVES

FUNCTIONAL

Information Processing
Administrators of Large School Systems

Alan Olkes (5/81-5/82)
Dade County Public Schools
13135 Coral Way
Miami, FL 33175
305/552-9894

Permanent Contact

Roy DuCharm
Prince Georges County
Public Schools
Upper Marlboro, MD 20870
301/952-4378

GEOGRAPHIC

Alaska

Norman Holthouse (6/81-6/82)
Anchorage School District
Pouch 6-614
Anchorage, Alaska 99502
907/269-2251

Alberta

Ann Brebner (7/81-6/82)
University of Calgary
2920 - 24th Ave. N. W.
Calgary, Alberta T2N 1N4
403/284-6493

Arizona

James Lee (5/80-4/82)
American Grad. Sch. Inter.
Management
Thunderbird Campus
Glendale, AZ 85306
602/978-5401

California Educational Data
Processing Association

Marilyn Carson (11/81-11/82)
Santa Clara Co. Office of Educ.
100 Skyport Drive
San Jose, CA 95110
408/947-6500

Florida

Mary Esther Raker (10/81-11/82)
Director Data Processing Hillsborough
P. O. Box 3408
Tampa. FL 33601

Randall Johns
Director Data Processing
1211 S. Mellonville Ave.
Sanford FL 32771
305/322-1252

Illinois

Lyle Smith (11/81-10/82)
Computer Science Dept. PM 566
Northern Illinois Univ.
DeKalb, IL 60115
815/753-0378

Iowa

Dorothy Creswell (11/81-10/82)
Norand Corporation
550 2nd St., S.E.
Cedar Rapids, IA 52401
319/366-7611

Winston Addis
Supt. of Schools
Mt. Vernon Comm. Schools
Mt. Vernon, IA 52314
319/895-8845

Manitoba

Lauran Sandals (6/81-5/82)
Educ. Psychology
University of Manitoba
Winnipeg, Manitoba R3T 3B3
204/474-9017

Edward Boorsma
Winnipeg SD #1
1577 Wall Street E
Winnipeg, MB R3E 2S5
204/775-0231

Maryland

Frank M. White (4/81-4/82)
Dept. of Comp. DP Tech.
Catonsville Comm. College
Baltimore, MD 21228
301/461-9321

Michigan

Lary Smith (10/81-10/82)
Wayne County ISD
33500 Van Born Road
Wayne, MI 48184
313/326-9300

Mid-America

James Wenger (4/82-3/83)
Emporia State Univ.
1200 Commercial
Emporia, KS 66801

Mid-South

Barbara Straka (3/82-2/84)
Dept. of Computer Science
Jackson State University
Jackson, MS 39217
601/968-2105

Minnesota

Linda Borry Hausmann (7/81-6/82)
1750 26th Ave. N. W.
St. Paul, MN 55112
612/631-2926

Nebraska

Donald Costello (6/81-5/82)
Univ. Nebraska Comp. Ctr.
3835 Holdrege St.
Lincoln, NE 68583
402/472-3763

New Jersey

Robert Osgoodby
Asbury Park Board of Education
1506 Park Ave.
Asbury Park, NJ 07712
201/775-0944

Irvin Gaydos
State Dept. of Education
225 West State
Trenton, NJ 08601
609/292-6037

New York

Donald Ross (10/81-10/82)
Ardsley High School
500 Farm Road
Ardsley, NY 10502
914/693-6300

North Carolina

John Beegle (3/81-3/82)
Western Carolina Univ.
P. O. Box 518
Cullowhee, NC 28723
704/227-7403

Ohio

Richard L. Acierto, (7/81-7/82)
Erie Co. Board of Education
2902 Columbus Ave.
Sandusky, OH 44807
419/627-7608

Oregon

Porter Sexton (1/82-12/82)
Portland Public Schools
501 N. Dixon
Portland, OR 97227
503/249-2000

Texas

Robert W. Hamilton (10/81-10/82)
Amarillo College
Box 447
Amarillo, TX 79178
806/376-5111

Virginia

Charles Allen (7/81-6/82)
VA Comm. College System
101 N. 14th St.
Richmond, VA 23219
804/225-2282

James Augustine
Norfolk City Pub. Sch.
800 E. City Hall Ave.
Norfolk, VA 23510
804/441-2458

Wisconsin

John Konopacky (7/81-6/82)
CESA 9
1927 Main St.
Green Bay, WI 54301
414/497-3690

Rolland Hicks
Dir. NW Comp. Support Ctr
CESA #6
West Park Ave.
Chippewa Falls, WI 54729

Geographic Affiliate Representative

Kenneth Hasledalen (5/81-5/82)
MN SD Data Proc. Joint Board
1925 W. County RD B2
St. Paul, MN 55113
612/633-9110

INSTITUTIONAL MEMBERS

Adams County School District #12, Northglenn, Colorado
Allegheny Intermediate Unit, Pittsburg, Pennsylvania
American Graduate School of International Management, Glendale, Arizona
Aramco Schools, Dhahran, Saudi Arabia
Baltimore City Public Schools, Baltimore, Maryland
Boulder Valley Public Schools, Boulder, Colorado
Broward County School Board, Ft. Lauderdale, Florida
Buffalo Public Schools, Buffalo, New York
Cincinnati Public Schools, Cincinnati, Ohio
Coast Community College District, Costa Mesa, California
College of DuPage, Glen Ellyn, Illinois
DODDS-Pacific Region Education Division, Seattle, Washington
Downers Grove High School North, Downers Grove, Illinois
East Texas State University, Commerce, Texas
Eastern Kentucky University, Richmond, Kentucky
Fairfax County Public Schools, Fairfax, Virginia
Fordham University, Bronx, New York

Hastings Public Schools, Hastings, Nebraska
Hillsborough County Public Schools, Tampa, Florida
Howard County Public School System, Ellicott City, Maryland
Hutchinson Community Junior College, Hutchinson, Kansas
International Institute of Applied Technology, Inc., Germantown, Maryland
Jackson Municipal Separate School District, Jackson, Mississippi
Jackson State University, Jackson, Mississippi
Jefferson County Public Schools, Lakewood, Colorado
Kansas City Kansas Community College, Kansas City, Kansas
Kansas State Department of Education, Topeka, Kansas
Lake Washington School, Kirkland, Washington
Mahoning County Office of Education, Youngstown, Ohio
Mankato State University, Mankato, Minnesota
Maricopa Community College, Phoenix, Arizona
Maryland State Department of Education, Baltimore, Maryland
Metro II, St. Paul, Minnesota
Miami-Dade Community College, Miami, Florida
Minnesota Educational Computing Consortium, St. Paul, Minnesota
Minnesota School District Data Processing Joint Board, St. Paul, Minnesota
New York State Education Dept., Albany, New York
Northwest Regional Education Laboratory, Portland, Oregon
Oklahoma City Public Schools, Oklahoma City, Oklahoma
Oral Roberts University, Tulsa, Oklahoma
Orleans Parish School Board, New Orleans, Louisiana
Pace University, New York, New York
Palm Beach County School Board, West Palm Beach, Florida
Palm Beach Junior College, Lake Worth, Florida
Rosemount Independent School District #196, Rosemount, Minnesota
Rutgers University, CCIS-CRC, New Brunswick, New Jersey
San Mateo County Schools, Redwood City, California
Shawnee Mission Public Schools, Shawnee Mission, Kansas
Tarrant County Junior College, Hurst, Texas
Thomas Jefferson University, Philadelphia, Pennsylvania
Topeka Unified School District 501, Topeka, Kansas
University of Alberta, Edmonton, Alberta, Canada
University of Delaware, Newark, Delaware
University of Iowa, Iowa City, Iowa
University of Pennsylvania, Philadelphia, Pennsylvania
U.S. Naval Academy, Annapolis, Maryland
Valencia Community College, Orlando, Florida
Wichita Public Schools, Wichita, Kansas

SUSTAINING MEMBERS

DATA GENERAL CORPORATION, Westboro, Massachusetts
DIGITAL EQUIPMENT CORPORATION, Marlboro, Massachusetts
HONEYWELL INFORMATION SYSTEMS, Waltham, Massachusetts
NATIONAL COMPUTER SYSTEMS, Minneapolis, Minnesota
TIME SHARE CORPORATION, Hanover, New Hampshire
WESTINGHOUSE DATASCORE SYSTEMS, Iowa City, Iowa

TABLE OF CONTENTS

INTRODUCTION

FORWARD i

ACKNOWLEDGEMENTS ii
  AEDS Board of Directors iii
  AEDS-82 Convention Team iii
  Affiliate Presidents and Representatives iv
  Institutional Members vi
  Sustaining Members vii

PAPERS

SESSION 1.2

MATCHING FUNCTIONS OF THE APPLE COMPUTER TO THE INSTRUCTIONAL NEEDS OF THE LEARNING DISABLED STUDENT
Archer, Doug 1

SOCIAL ISSUES IN COMPUTING: A COURSE FOR COMPUTER SCIENCE MAJORS
Bailey, T. E. 5

IDENTIFICATION AND EVALUATION OF EDUCATIONAL APPLICATIONS SOFTWARE
Bockmann, Fred 9

INTERACTIVE LANGUAGE EXPERIENCES BETWEEN THE COMPUTER AND THE STUDENT
Kuchinskas, Gloria 12

DESIGNING INSERVICE COMPUTER LITERACY TRAINING FOR TEACHERS
Martin, C. Dianne and Heller, Rachelle S. 15

IMPROVING THE VALIDITY OF COMPUTERIZED MULTIPLE-CHOICE TESTS IN MATHEMATICS
McHenry, Dr. Hugh 19

INSTRUCTIONAL USES OF COMPUTERS: THE GOOD, THE BAD, AND THE UGLY
Sloan, Sally A. 23

SESSION 1.2 (Continued)

TEACHING YOUNG CHILDREN TO PROGRAM IN BASIC
VanHorn, Royal 28

SESSION 1.3

MICROCOMPUTER INSTRUCTIONAL SUPPORT IN THE CLASSROOM (MISIC) PROJECT - MICROCOMPUTERS IN THE CLASSROOM K-12 A "HOW TO DO IT" REPORT
Becht, Paul and Becht, Sara-Maria F. 32

CAI: PRACTICE, ART, AND SOMETIMES SCIENCE
Burke, Robert L., Ed.D. 34

A COMPUTER LITERACY PROGRAM FOR ELEMENTARY AND MIDDLE SCHOOL CHILDREN AND TEACHERS
Halapin, Jack 38

USING THE MICROCOMPUTER TO APPLY DATA-BASED DECISION RULES FOR ASSISTING IN THE SELECTION OF INSTRUCTIONAL STRATEGIES FOR THE HANDICAPPED
Hasselbring, Ted S. and Hamlett, Carol L. 43

A PROBLEM SOLVING AUTHORING SYSTEM
Perry, Ted 47

COURSEWARE CRITERIA FROM AN INSTRUCTIONAL DESIGN PERSPECTIVE
Roblyer, M. D. 51

AND THE TWAIN SHALL MEET: COMPUTERS AND THE TEACHING OF ENGLISH
Shostak, Robert 55

DISTRIBUTED DATA PROCESSING: GENERAL CONSIDERATIONS AND A USER EXPERIENCE
Umar, Amjad 59

CRITERIA FOR ESTABLISHING A COMPREHENSIVE COMPUTER PROGRAM FOR A SMALL LIBERAL ARTS COLLEGE
Wilde, Jane H. 71

SUCCESSFUL IMPLEMENTATION OF COMPUTING TECHNOLOGY IN SCHOOLS THROUGH TEACHER EDUCATION, PROFESSIONAL DEVELOPMENT AND SUPPORT
Willis, Cam 75

EDUCATIONAL COMPUTING: SUPPORT FUNCTIONS AND STUDENT OBJECTIVES
Bybee, Dennis L., Ph.D. 80

FLORIDA CENTER FOR INSTRUCTIONAL COMPUTING
Engel, Dr. C. William 83

SELECTING EDUCATIONAL APPLICATIONS FOR MICROCOMPUTERS
Hausmann, Linda Borry 88

MICROCOMPUTERIZED KAUFMAN WISC-R HYPOTHESIS GENERATION
Ingram, Rich 92

SESSION 1.4 (Continued)

MICROCOMPUTERS: ADMINISTRATIVE TIMESAVERS
Kehrer, Gary A. and Schepia, Nick A. 94

FLORIDA'S INFORMATION RESOURCE NETWORK
Olkes, Alan T. 98

IT DOES NOT HAVE TO BE A BLACK BOX: THE CREATIVE USE OF A MICROCOMPUTER AS A COMPUTATIONAL AID IN THE EDUCATIONAL ENVIRONMENT
Pas, Eric I. and Anapol, Edward 102

LEVELING COURSES FOR STUDENTS PURSUING A MASTERS DEGREE IN COMPUTER SCIENCE
Swigger, Dr. Katheleen M. and Zamecnik, Dr. Fred J. 106

THE DESIGN, DEVELOPMENT AND EVALUATION OF A LOW-COST COMPUTER MANAGED SPELLING SYSTEM
Thomas, Rex and Gustafson, Brian 110

WORD PROCESSING/TEXT EDITING USING MICROCOMPUTERS
Wetterlind, Dr. Peter 115

SESSION 1.5

COMPUTER LITERACY TRAINING NEEDS OF INSERVICE AND PRESERVICE TEACHERS: IMPLICATIONS FOR TEACHER EDUCATION PROGRAMS
Anderson, Cheryl 119

CALCULUS TUTORIAL SYSTEM
Arganski, Nicholas 123

HIGHER EDUCATION'S RESPONSIBILITY TO JOB TURNOVER IN COMPUTER-RELATED JOBS
Bailey, M. Gene and Black, David 127

MICROCOMPUTERS IN THE GIFTED/TALENTED PROGRAM
Berry, Mary 131

EFFECTIVENESS OF COMPUTER-BASED INSTRUCTION IN JOB CORPS
Geller, Daniel M., Ph.D. 134

MICROCOMPUTER SOFTWARE: EVALUATE--EVALUATE--EVALUATE
Taber, Florence 138

STUDY TO EXAMINE COMPUTER TECHNOLOGY FOR IMPLEMENTATION INTO A COLLEGE READING PROGRAM
Williams, Carmelita 140

SESSION 1.6

INSTRUCTIONAL MANAGEMENT: CIMS-IMPACT CASE STUDY
Adams, Charles and Cameron, Kathryn 143

SESSION 1.6 (Continued)

DEVELOPMENT OF A STATEWIDE SYSTEMS ARCHITECTURE: EXPERIENCES FOR THE STATE OF MINNESOTA
Bahn, Theodore 147

THE IMPACT OF MICROCOMPUTING ON SECONDARY SCHOOL ADMINISTRATORS
Bowers, Larry 152

COMPUTER SUPPORTED STUDENT MANAGEMENT FOR A LARGE CLASS
Boyle, Thomas A. 156

DEVELOPING A FACULTY COMPUTER LITERACY PROJECT IN A MULTI-CAMPUS COMMUNITY COLLEGE
Guthrie-Morse, Barbara, Ph.D. 160

COGNITIVE STYLE AND A FIRST COURSE IN COMPUTER SCIENCE: A SUCCESS STORY
Hassell, Johnette 163

THE ABC'S OF FARMING OR AGRICULTURE BY COMPUTER
Hintz, Thomas R. 167

THE APPLICATION OF DISTRIBUTED PROCESSING AND NETWORKING AT BROWARD COUNTY SCHOOLS TO MANAGE STUDENT RECORDS AND PERFORM INSTRUCTIONAL MANAGEMENT
Kauffman, Fred 171

REINFORCING READING SKILLS USING THE MICROCOMPUTER AND SPECIALIZED SOFTWARE DESIGN BY ELEMENTARY AND MIDDLE SCHOOL TEACHERS
Lagotic, Diana, Ph.D; Losch, Lorraine; Domenech, Lynn 176

EVALUATION RESEARCH - A CATALYST FOR CHANGE IN COMPUTING
Lee, L. D. and Haughey, D. G. 178

STOREROOMS OR CLASSROOMS? THE FUTURE OF MICROCOMPUTERS IS IN THE TRAINING OF ELEMENTARY SCHOOL TEACHERS
Richard, Dr. Tom 184

SESSION 2.1

THE NUTS AND BOLTS OF SELECTING AND MAXIMIZING COMPUTER ASSISTED INSTRUCTION
Conkling, Richard D. 188

APPLYING COMPUTERS EFFECTIVELY IN THE SMALL TWO-YEAR COLLEGE
Conrad, Lora P. 192

INTEGRATING CAI AND VIDEOTAPE FOR SCIENCE AND MATHEMATICS
Ignatz, Mila E. 196

ALGORITHMS AND THEIR SOURCE CODE IMPLEMENTATION
Nuesse, David A. 198

MATCHING INSTRUCTIONAL PROGRAMS TO LEARNING PROBLEMS
Pollard, James, Ed.D. 203

SESSION 2.1 (Continued)

A MICROCOMPUTER-BASED INSTRUCTIONAL MANAGEMENT SYSTEM
Rodel, Lee J. 206

WHAT IS REAL COURSEWARE?
Rohm, Robert R. 210

TOWARD A FUNCTIONAL INTEGRATION OF COMPUTERS IN THE EDUCATIONAL PROCESS: A MICROCOMPUTER APPLICATION FOR FACILITATING MEMORIZATION
Schilmoeller, Kathryn J; King, Dennis R.; Schilmoeller, Gary L. 213

CHANGE-LIKE IT OR NOT!
Wallisch, Lt. Col. (Dr.) Bill 217

SESSION 2.2

UNDERSTANDING THE "CAI PHENOMENON"
Bright, George W. 221

MICROCOMPUTER APPLICATIONS IN THE COMMUNITY COLLEGE
Dellow, Dr. Donald A. and Poole, Dr. Lawrence H. 225

PROCEDURES FOR DEVELOPING ADMINISTRATIVE SOFTWARE
Loewen, Duane 229

THE USE OF GEOPROCESSING IN EDUCATIONAL RESEARCH AND DECISION SUPPORT
Sexton, Porter 232

SESSION 2.3

COST EFFECTIVE COMPUTER GRAPHICS TERMINAL UTILIZATION: THE STUDENT AS TEACHER
Brillhart, L. V. and Bell, Eric A. 236

CREATIVE PROGRAMMING FOR YOUNG MINDS
Buxton, Marilyn and Taitt, Henry A. 241

MICROCOMPUTER PATIENT EDUCATION COURSEWARE
Canine, Kent, Ph.D. and McWeeny, Mae 244

MICROCOMPUTERS IN COOPERATIVE GROUPS
Corens, Ken and Maggie 247

MICROCOMPUTERS IN SPECIAL EDUCATION--BOON OR BANE?
Hannaford, Alonzo E. 250

TRANSPARENT SORTS BETTER SERVE USERS
Muldrow, William R. 254

INTEGRATING MICROCOMPUTERS INTO THE MATHEMATICS CURRICULUM--USING THE COMPUTER AS AN ADJUNCT INSTRUCTOR
Schwandt, Lynn C. 258

MANAGING A MAJOR RESOURCE - MANAGEMENT AND INSTRUCTIONAL INFORMATION
Somers, Brenda 261

SESSION 2.3 (Continued)

COMPUTER LITERACY IN THE ELEMENTARY SCHOOL: AN ARGUMENT FOR CHANGE FROM WITHIN
Winner, Alice Ann 265

SESSION 2.4

WHAT SHOULD BE INCLUDED IN COMPUTER LITERACY TRAINING FOR TEACHERS?
Andrews, Bernice Doerr 269

PROJECT CAISH: CAI AND SUPPORT FOR THE HANDICAPPED
Brown, Warren 273

DECENTRALIZING SCHOOL ADMINISTRATIVE SUPPORT SERVICES WITH SMALL BUSINESS MICROCOMPUTERS IN A LARGE SCHOOL SYSTEM
Bybee, Dennis L., Ph.D. 277

USING COMPUTER GRAPHICS AS A CRITICAL COMPONENT IN MATHEMATICS INSTRUCTION
Flake, Janice L. 279

COMPUTER-ASSISTED TEACHING OF ARABIC
Johnson, J. Matthew and Bonney, William W., Jr. 283

TECHNIQUES FOR USING MICROCOMPUTERS TO AID LEARNING DISABLED AND MENTALLY RETARDED STUDENTS
Mason, Marguerite 286

COMPUTER LITERACY - TEACHER EXPECTATIONS AND STUDENT ACHIEVEMENTS
McCabe, Jack 290

COMPUTER LITERACY FOR EDUCATIONAL ADMINISTRATORS
Spero, Samuel and Jonas, Jan 292

EVALUATION OF THE IMPLEMENTATION AND MANAGEMENT OF MICROCOMPUTERS FOR INSTRUCTION
Van Dusseldorp, Ralph 296

SESSION 3.1

FULFILL INSTRUCTIONAL COMPUTING NEEDS WITH VARIOUS MICROCOMPUTERS
Brumbaugh, Dr. Kenneth E. 300

COMPREHENSIVE DATA MANAGEMENT SYSTEM (CDMS) FOR INDIVIDUAL EDUCATION PLANS AND SPECIAL EDUCATION PROGRAM MANAGEMENT
Guth, Robert O., Ed.D. 304

TAXONOMY OF INSTRUCTIONAL STRATEGIES FOR COMPUTER BASED EDUCATION
Hall, Keith A.; Comer, Ronald C.; Merrill, John A. 307

EVALUATING AND SELECTING A BUSINESS BASIC
Hoey, James M. 311

SESSION 3.1 (Continued)

OBSERVATIONS ON THE DYNAMICS OF K-12 TEACHER INVOLVEMENT WITH MICROCOMPUTERS
Judd, Dorothy H., Ed.D. and Judd, Robert C. 315

MANAGING INFORMATION FOR NEW KINDS OF LEARNERS
Markovich, Greg and Malling, Joan 319

COMPUTER LITERACY EDUCATION FOR K-6 TEACHERS
Philipp, Charles C.; Muntner, Judie; Cutlip, Patricia 322

FACTORS RELATING TO THE UTILIZATION OF INFORMATION BY SUPERINTENDENTS FOR DECISION-MAKING
Somers, Brenda 326

SESSION 3.2

APPLYING INSTRUCTIONAL AND COMPUTER TECHNOLOGY TO THE DEVELOPMENT OF IEP'S: PROJECT RECIPE
Reid, George A., Jr. and Bell, H. Sanders, III 330

STUDENT SELF-TRACKING USING A MICROCOMPUTER
Strong, Mary Winifred 334

BASICS, NOT BASIC: OR TEACHING WRITING TO THE COMPUTER
Stuart, Dr. Ann 337

ESTABLISHING A HARD-DISK BASED MICROCOMPUTER NETWORK
Tinsley, Jack T. 339

TO COMPUTERIZE OR NOT? A QUESTION OF GIGO!!
Zuckerman, Robert A., Ed.D. 342

SESSION 3.3

THE DESIGN OF A STRUCTURED CAI LANGUAGE AND CAI SUPPORT SYSTEM
Harris, Marshall 346

COMPUTER POWER: A COORDINATED SERIES OF HIGH SCHOOL CURRICULA
Hughes, Charles E. and Moshell, J. Michael 350

THE POLITICAL PLATFORM OF THE COMPUTER
Leslie, Judith W. 356

A SUPERINTENDENT'S PERSPECTIVE ON MANAGEMENT INFORMATION SYSTEMS
McFatter, William 360

MICROCOMPUTERS: GETTING STARTED IN A JUNIOR HIGH SCHOOL
Pinski, Mary M. 364

LONG RANGE PLANNING FOR COMPUTER USE - A TASK FORCE MODEL
Raucher, S. M. and Koehler, T. J. 367

USING MICROCOMPUTERS TO IMPROVE YOUR HEARING IMPAIRED PROGRAM
Sarapata, Susan L. 374

SESSION 3.3B

OCCUPATIONAL PROFICIENCY PREPAREDNESS OR FUTURE SHOCK?
Adams, Sue 377

LEFT-HANDED LEARNERS, THEIR HEMISPHERICITY PROBLEMS AND COMPUTERS
Donoian, George, Ed.D. and Donoian, Eleanor, BA 381

BIG PROBLEM - STUDENT SCHEDULING AND GRADE REPORTING, BIG SOLUTION - THE MICROCOMPUTER
Mosow, Dr. David K.; Bolton, Brenda; Young, Henderson 385

APPLIED DISCRIMINANT MODEL FOR INFORMATION SYSTEMS EVALUATION
Rushinek, Avi; Rushinek, Sara F.; Stutz, Joel D. 389

THE COMPUTER IN THE SCHOOLS -- WHAT'S BEEN SAID TO THE TEACHER?
Saltz, Martin 393

FLORIDA CHOICES - HELPING STUDENTS CHOOSE
Thomas, Glenn and Wooley, William 396

MICROCOMPUTER APPLICATIONS IN THE SPECIAL EDUCATION CLASSROOM
Turner, Greg 400

THE DANGER OF THE REDUNDANT MICRO
Vinlove, Donald J. 403

SESSION 3.4

IMPLEMENTING INSTRUCTIONAL COMPUTING AND COMPUTER LITERACY IN A SCHOOL OR COLLEGE
Bell, Frederick H. 406

ADDRESSING THE NEED FOR PRE-COLLEGE TEACHER TRAINING IN COMPUTER SCIENCE IN THE STATE OF GEORGIA
Clark, Dr. Frank C. and Carter, Dr. Harrison S. 412

EFFECTS OF PROGRAMMING ON MATHEMATICS ACHIEVEMENT
Harrison, Mary O. 416

COMPUTER SIMULATION LABORATORY FOR DESIGN OF EXPERIMENTS
Hirsch, R. A.; Geremia, J. O.; Wu, C. 421

BUILDING AN EFFICIENT AND EFFECTIVE MANAGEMENT INFORMATION SYSTEM
Marshall, Jon C. 428

OUR ADVENTURES WITH THE COMPUTER
McClellan, Jane and Griffin, Holley 432

COMPUTER EDUCATION - A PRELIMINARY PLAN OF ACTION FOR THE DADE COUNTY PUBLIC SCHOOLS
Olkes, Alan T. 436

USING A MICROCOMPUTER TO MANAGE LARGE FILES OF INSTRUCTIONAL PICTURES
Price, Samuel T. and Dillingham, Larry M. 440

SESSION 3.5

COMPUTER SIMULATIONS STIMULATE INDUCTIVE THINKING
Gress, Eileen K. 445

VOCATIONAL TRAINING FOR HIGH SCHOOLS IN THE AGE OF THE MICRO
Lorton, Paul, Jr. and Muscat, Eugene J. 450

SOFTWARE SHARING IN FLORIDA COMMUNITY COLLEGE BURROUGHS USERS CONSORTIUM
Monish, A. R. 453

AN ON-LINE LIBRARY CIRCULATION SYSTEM
Pennington, Billy 457

THE COMPUTER: AN AFFECTIVE MACHINE
Roser, Jeff 461

COMPUTERS+BOOKS+KIDS=REAL LEARNING
Wall, Elizabeth S. 465

COMPUTER-MORPHOSIS
Weaver, Sister Catherine, O.S.F. 469

SESSION 4.1

SCIENCE LITERACY IN THE PUBLIC LIBRARY
Bork, Alfred; Franklin, Stephen; Von Blum, Ruth; Trowbridge, David; Kuntz, Barry L. 473

COMPUTING FOR THE HANDICAPPED
Hazan, Paul L. 476

INDEX

ALPHABETICAL BY TITLE 479

ALPHABETICAL BY AUTHOR 487

MATCHING FUNCTIONS OF THE APPLE COMPUTER TO THE INSTRUCTIONAL NEEDS OF THE LEARNING DISABLED STUDENT

-By Doug Archer, Special Ed. Computer Specialist

There is little in the history of education to compare with the recent fervor that grips K-12 teachers throughout North America regarding the instructional use of the microcomputer. In most states the growth in numbers of units has been phenomenal. In Iowa, not generally known for jumping readily on band wagons of innovations, the number of Micros has grown over 300% in the last year. In the fall of 1980, an informal survey showed just over 300 micros, of all kinds in use, but, when school began in the fall of 1981 there were over 1573 microcomputers in use in the state's schools. With a projected 11 million dollars to be spent on computers in Iowa over the next year, there is every reason to expect this rate of growth will continue, or even to increase. All this despite considerable cuts in education budgets, over the past two years.

In Area 14, an eight county region, in Southwest Iowa, containing two of the state's poorest counties, and many of the smallest schools, this same growth rate has been constant for the past two years. Of even further interest is that 75% of Area 14's microcomputers are located in Resource Rooms, being used regularly by students functioning at levels from mild learning disabled to moderately handicapped.

Although for many involved in Computer Assisted Instruction (CAI), the Special Education population may not immediately come to mind as appropriate to fully utilize the computer's many capabilities, closer review will reveal some exciting possibilities. To compare the teaching methods used to reach the majority of instructional objectives being worked on in Special Education with the capabilities of the computer is to outline a match made in Heaven.

That computers have been received by Special Education teachers with such enthusiasm, will come as no surprise to one familiar with that student population. In few other instructional areas, can a greater need be found for the things a computer can do. Perhaps that is why much of the significant research that has been completed in the last five years, evaluating the validity of computer assisted instruction has been done with handicapped learners. Although until the last two years, minicomputers were the delivery device, microcomputers, such as the Apple and the TRS-80, are rapidly becoming the format.

In Area 14, microcomputers are being used in Special Education in two main areas: data management and as a delivery device for instruction.

One of the major complaints about the legislation (Public Law 94:142) that dictates how students in Special Education are evaluated and placed in instructional programs has been the large amount of paperwork, duplication, and documentation involved. There are multiple copies of everything, stored in files in several locations. This is further complicated by the team planning approach to program development, which perform various diagnostic tests, the results of which must be compiled, compared, and interpreted. The end result of all this is to be a coherent instructional plan (known as the IEP or Individualized Education Plan) that can be clearly evaluated, easily monitored, and carried out by everyone involved.

Boiled down to its simplest components the IEP is data compilation, data analysis, and data management. In short the computer is designed to do well, much of what Special Educators need to do the most.

Presently several software data management programs are being marketed to meet this need. Some are general data management programs, some are custom designed, specifically for the IEP.

Green Valley Area Education Agency, in Creston, Iowa, is currently completing a customized version of the Programming for Individualized Education (PIE) program, developed by Dr. Michael Athey. Dr. Athey, a former school psychologist, applied much of his experience working with P.L. 94:142 to the design of this program. Although originally designed for the North Star computer, Green Valley's PIE will work on a 48K Apple II.

The operation of the PIE program is fairly straight forward, and is designed for the computer novice teacher. After setting up demographic student data files, the teacher indicates the area of instructional need, as determined by the diagnostic testing. The computer then matches this area with all appropriate instructional objectives. After selecting those appropriate for the student the teacher can then match each objective with the appropriate instructional material. The results of these searches, becomes the student's IEP. The program diskette is designed to hold 30 student files of such data.

Such a search will require large numbers of objectives and materials, to assure the most appropriate choices for the student. To permit this the Green Valley system includes a 33 MegaByte hard disk. The present plan will allow area teachers to set up student files, entering areas of need, and then either run a search on the agency's objective and materials bank via phone modem, or by sending their program disk, to the agency's central office, and having Green Valley's staff run searches for them. The results could then be loaded onto the student disk, and returned to the school via a twice weekly van delivery. Thus each building level teacher could maintain all student IEP's with a single disk drive and one disk.

In addition to developing and storing the IEP, PIE performs some other quite useful functions, including procedural tracking of each student, noting who has done what, and when. There is even allowance for observational data.

The advantages of such an electronic planning device include:

1) Documentation, whenever desired, without the duplication of numerous paper files.

2) Ease of data duplication and transportation.

3) Ease of rewriting or updating records. This will be especially noticeable with those students that move from district to district.

4) Ability to rapidly compile student statistics for government reporting or program evaluation.

5) Frees up considerable block of teacher time, previously spent on non-instructional activities, while at the same time greatly increasing accountability and documentation.

6) Automates the mechanical part of the instructional planning process, greatly dimminishing the time between initial referral to final program placement.

The other application widely used by Special Education teachers would come loosely under the heading of using the Apple as a delivery device for instruction.

While using a computer as a teaching machine is far from new, mainframe versions having been used since the early fifties, the availability of the microcomputer has sparked a rebirth in the whole idea of students directly interacting with the machine.

Although the common formats of CAI presentation are used, including 1) Drill and Practice, 2) Tutorial, and 3) Testing, these must take on a much more precise dimension, when used with the learning disabled.

Since the key ingredient to most instructional methods with this population is structure, the computer programs used, must take advantage of the multiple levels of activity and the speed of the computer. This is true of both commercially available courseware and authoring systems designed to help teachers assemble their own individualized lessons. Management, may be either by the computer - as in self contained curriculum (where all the lessons, as well as student performance is stored by the program); or by the teacher, using score sheets, learning packets, or other semi-independent methods. It is important to note that even though the micro may be a new way of delivering the instruction content, it can fit easily into the existing curriculum, and into most instructional models. Using a computer does not necessarily mean dismantling your classroom.

If computer assisted instruction is to become a viable mode of instruction, it must prove to be both instructionally valid and cost effective. The best way to achieve this is to do something that frees the teacher from a repetitious, mechanical task, so that he or she can do those things that only the teacher can do. Thus the amount of success in using a computer in an instructional setting is directly proportionate to the appropriateness of the method to the learner, and the increase in instructional effectiveness for the teacher.

This has been especially useful for itinerant teachers, serving several students spread over a sparsely populated region. Using the computer these teachers can supply and manage individualized lessons for most of their students. Since open formated tutorial programs are available, all the teacher (or in this case a secretary) must do to prepare a lesson is to put in the pertinent questions, answers, and parameter controls (such as how many attempts, etc.). In doing so the number of repetitions is dramatically increased (from 2 times to 50 times a week), while actually reducing their preparation time, since most lessons have multiple users. By accessing the teacher file part of the program, student progress may be monitored and further assignments made.

In addition to the prepared tutorial and drill format software, there is a growing number of so-called "authoring systems." These programs are designed to allow the non-programmer to easily put together an individualized lesson, often with substantial parameter controls, For the past year Green Valley has been developing programs using a teacher utility known as the BLOCKS authoring system, developed by the Kyde-Tyme Project in Sacramento, California. Although originally developed for teachers of hearing impaired students, the program can be used to develop CAI lessons for any student. The program boasts a graphics library of 650 graphics that the teacher loads into the student lesson file using the Apple game pods. These are placed anywhere on the screen desired, and there is no limit to how many can be placed in each frame. After selecting the graphic desired for each frame, the question and acceptable answer are entered. The parameters of the answer are quite flexible, and can be as general or as specific as the teacher desires. There is also a fully developed management system to store student progress for teacher review, as well as a utility program to add more graphics using a Graphic Tablet.

SOCIAL ISSUES IN COMPUTING: A COURSE FOR COMPUTER SCIENCE MAJORS

T. E. Bailey

Department of Administrative Sciences
University of Central Arkansas
Conway, Arkansas 72032

(This paper reports the results of a project undertaken while the author was with the Department of Computing and Information Sciences at the Oklahoma State University, Stillwater.)

A course based on "Computers and Society" in ACM's Curriculum '78 has been developed. Differing views from computer scientists, science fiction writers, social scientists, humanists, philosophers, and government spokesmen are presented, discussed, and analyzed. A perspective on the evolution of computing and its past, present, and future consequences is developed. Discussion is not limited to the United States, but also includes problems and perspectives in other cultures--West, East, and Third World. In addition, students work in groups, researching approved supplementary topics and presenting their results to the class.

In ACM's Curriculum '78 there appears a course titled "Computers and Society" (1). The stated objectives are

> to present concepts of social value and valuations;
> to introduce models which describe the impact of computers on society;
> to provide a framework for professional activity that involves explicit consideration of and decisions concerning social impact;
> to present tools and techniques which are applicable to problems posed by the social impact of computers.

Such a course was developed at Oklahoma State University and placed into the required core for computer science majors.

Rationale

Computer scientists, in their enthusiasm for computers and their uses, can easily lose sight of possible effects of their systems upon individuals and society. Since computers affect public policy, the decision-making function of governments and large organizations, and man's view of himself and the world, the seriousness of the consequences which may accrue from the actions of computer scientists requires that they be people of professional, social, and personal maturity. They must be able to ask human questions regarding what they are doing and why they are doing it. Joseph Weizenbaum (3) has said,

> We can count, but we are rapidly forgetting how to say what is worth counting and why.

Kenneth Peter, in a collection of essays called The Hacker Papers, wrote:

> Weak-willed people, people with unstable social lives, people in formative stages of their lives, should not become involved in computer science. It should be left until they are truly able to make decisions and be aware of all the consequences.

The computer science student must

be encouraged to develop a broadened perspective, an awareness of how his actions as a computer scientist may impact upon society and individuals. He must grasp the power of information and those who process it. Otherwise, he may fall victim to a serious trap.

> He may so thoroughly commit himself to what he naively perceives to be computer science, that is, to the mere polishing of his programming skills, that he may effectively preclude studying anything substantive. . . . When such students have completed their studies, they are rather like people who have somehow become eloquent in some foreign language, but who, when they attempt to write something in that language, find they have literally nothing of their own to say (4).

Those who design and teach the computer science curriculum must not be guilty of what Frank Herbert (6) calls "mind-slaving." Speaking of a teacher of mathematics, he said,

> . . . he was a mind-slaver and his enslaving process could be understood with extreme simplicity: he transferred technical knowledge without a transfer of values.

Weizenbaum (5) insists that the teacher must

> . . . affirm that the computer is a powerful metaphor for helping us to understand many aspects of the world, but that it enslaves the mind that has no other metaphors and few other resources to call on . . . .

Any responsible computer science program must instill an awareness of actual and potential social consequences of computer applications into students in all courses. Teachers must include concepts of social and professional responsibility along with other instructional materials. An upper-level course, in which human and social questions related to computing are investigated in depth, should be required as a capstone for the curriculum. The other courses deal with doing (what a computer scientist is), while the capstone course deals with being (who a computer scientist is).

Prerequisites and Structure

Three prerequisites--junior standing, 9 credit hours of computer science, and an English course in advanced technical writing--were imposed so that students would have greater breadth and maturity. The English course (required of all majors anyway) is prerequisite because students must write essay tests and a research paper.

Students who do not meet the computer science requirement, however, are encouraged to enroll in the course, particularly students from social disciplines and humanities. Curriculum '78 endorses participation of non-majors:

> A mixture of majors in such a course would provide broadening interchange and would benefit both the computer science students and the other majors.

Activities in the course fall into four categories: reading selections from the literature, classroom presentation by the instructor of additional material, active classroom discussion, and a written research report which is also presented orally in class.

There is a wealth of literature. The problem is getting a representative sampling and having it in a form readily accessible to students. A good anthology and/or a collection of paperback books could be used. We chose an anthology, Adams and Haden's Social Effects of Computer Use and Misuse (7), and a paperback book, Weizenbaum's Computer Power and Human Reason (2). Adams and Haden have a good collection of stimulating and controversial articles. Weizenbaum was selected because he is controversial, quotes heavily from a number of sources, and raises some significant questions. (Whether one agrees with his answers is inconsequential; the questions are worthy.)

The instructor reviewed and discussed additional articles and authors in class. Included in this group, among others, are the Privacy Act of 1974, the proposed Computer Crime Bill (8), "The Vulnerable Computer Society" from Sweden's Ministry of Defense (1976), and "IBM's Guidelines to Employee Privacy" (9).

Class discussion is fairly easy to elicit, since many of the topics and ideas are at best controversial and at worst shocking. The instructor must take care to keep the focus on the issues and to keep a balance by presenting alternatives, arguments, and counter arguments. There must be questioning and critical examination. A grasp of what questions are important and why is at least as important as answers.

Students were randomly divided into groups of three for the research project. Each group chose a topic, which had to be approved by the instructor, insuring substance and preventing duplication. The group prepared a written report discussing the background and present state of the topic and its future implications. The group also presented the results orally to the class and responded to questions.

Testing and Grading

Testing consisted of three essay tests and a final essay test. A week before each regular test, the students received a set of questions for which they were to prepare responses. On the day of the test, they were given a subset of the questions to write in class. The final test consisted of a subset of questions from previous questions and a few new questions (also given in advance). Students were allowed to use notes, textbooks, and dictionaries on all tests. Responses to test questions were evaluated on whether they were thorough, logical, reasonable, well stated, and focused on the issues.

Students were told at the outset that one purpose of the course was to provide opportunity to practice communication skills and that they would be graded in part on English. After the first test, most students took this challenge seriously and wrote papers of fairly high to very high quality.

Half of the final grade was based on tests and half on the research project, divided half and half between the written and the oral reports. On the research project, each group had the choice of being graded as a group or as individuals.

Results

"Social Issues in Computing" was first offered in its present form during the Spring Semester of 1981. From all appearances, it was a successful experience overall. A few students seemed unconvinced and inconvincible of the importance of the course and the issues raised. As a whole, however, students responded well and performed above the instructor's expectations.

The course is very demanding on the instructor. A team teaching approach should be tried. This would not only reduce the instructional and grading load, but would also provide broader experiences for the class.

There is one final observation of interest. Computer science students have developed thinking oriented toward predictable, precise results. "Social Issues in Computing" requires an entirely different, essentially opposite way of thinking. Some students find the transition difficult, if not impossible. But the real significance of the course lies in this experience. After all, human beings are not computers; they are wonderfully unpredictable and imprecise--most of all, wonderful. Abbe Mowshowitz, himself a computer scientist, has said, Human superiority consists in the ability to act irrationally (10).

References

1. Communications of the ACM, Vol. 22, No. 3, pp. 155-6 (1979).

2. Weizenbaum, Joseph. Computer Power and Human Reason. San Francisco: Freeman, 1976.

3. Ibid., p. 16.

4. Ibid., pp. 277-8.

5. Ibid., p. 277.

6. Herbert, Frank. Children of Dune. New York: Berkley, 1976, p. 198.

7. Adams and Haden. Social Effects of Computer Use and Misuse. New York: Wiley, 1976.

8. S.1766, 95th Congress (1977).

9. Harvard Business Review, September-October, 1976, pp. 82-90.

10. Mowshowitz, Abbe. Inside Information. Reading: Addison-Wesley, 1977, p. 42.

Selected Bibliography

Baer, Robert M. The Digital Villain. Reading, Mass.: Addison-Wesley, 1972.

Ellul, Jacques. The Technological Society. New York: Alfred A. Knopf, 1964.

Gotlieb, C. C., and Borodine, A. Social Issues in Computing. New York: Academic Press, 1973.

Mathews, Walter M., editor. Monster or Messiah? The Computer's Impact on Society. Jackson: University Press of Mississippi.

McCorduck, Pamela. Machines Who Think. San Francisco: Freeman, 1979.

Mowshowitz, Abbe. The Conquest of Will. Reading, Mass.: Addison-Wesley,

Mumford, Lewis. The Myth and the Machine: Technics and Human Development. New York: Harcourt Brace Jovanovich, 1967.

Mumford, Lewis. The Myth and the Machine: The Pentagon of Power. New York: Harcourt Brace Jovanovich, 1970.

Raphael, Bertram. The Thinking Computer. San Francisco: Freeman, 1976.

Roszak, Theodore. The Making of a Counter-Culture. New York: Doubleday, 1969.

Servan-Schreiber, Jean-Jacques. The World Challenge. New York: Simon and Schuster, 1980.

Teague, Robert, and Erickson, Clint. Computers and Society: A Reader. St. Paul: West, 1974.

Toffler, Alvin. Future Shock. New York: Random House, 1970.

Van Tassel, Dennie. The Compleat Computer. Palo Alto: SRA, 1976.

Wessel, Milton R. Freedom's Edge. Reading, Mass.: Addison-Wesley, 1975.

Westin, Alan F., and Baker, Michael A. Databanks in a Free Society. New York: Quadrangle Books, 1972.

IDENTIFICATION AND EVALUATION OF EDUCATIONAL APPLICATIONS SOFTWARE

The past few years have seen an explosion in the development of intelligent machines. The microcomputer, unheard of until the mid 1970's, is now found in many schools across the country. Technology continues to speed forward with the development of more powerful computers that are easier to use. Science fiction is becoming science fact at a rapidly increasing pace.

The development of suitable software to run on these increasingly prevalent computers has lagged significantly behind the development of the hardware. The computers found in schools are being under-utilized at best, and barely used in many cases. The cause of this situation is the difficulty of finding quality software with adequate support materials for the teacher.

Anyone can become a software author- and unfortunately many people untrained in curriculum design have become authors of educational software. There exists a vast sea of poorly designed software with occasional islands of excellent software without support materials and rare prominences of high quality software with usable and complete support materials.

One of the objectives of MICRO-IDEAS is to identify those pieces of software that not only are well executed in terms of programming and curricular design, but that also have sufficient support materials to make an adequate teachers guide that will facilitate classroom use of the software.

There are three areas of concern in evaluating a piece of software for classroom use- the educational pedagogy the piece is based on, the level of technical expertise used in creating the software, and the sophistication of human interfacing used in the execution of the program. Simply put, does the program do something educationally worthwhile and appropriate, does the program make use of the unique capabilities of the microcomputer in a polished and professional manner, and is the program easy to use and hard to fool?

EDUCATIONAL PEDAGOGY

All to often, educational professionals forget everything they know about the development of solid curriculum when a new technology is introduced. That which was educationally sound prior to the introduction of microcomputers is still valid today. A piece of software needs to meet the same criteria that would be applied to textbook, filmstrip, or movie. The computer does not make bad education suddenly worthwhile- it just presents junk in a more appealing manner.

The criteria of logical development, appropriate modes of presentation, and reinforcement of correct responses that apply to non-computerized teaching should be applied to the evaluation of software. A good lesson is a good lesson, regardless of the way it is presented.

TECHNICAL EXPERTISE

A movie that suffered from poor directing, poor acting, poor setting, or poor plot would not be recommended for educational use. The same is true of computer software.

Microcomputers present similar problems to the software producer. The piece of software must have a polished, well executed appearance. Such unacceptable flaws as words "broken" at the end of a line, misspellings, and improper grammar are all too prevalent in many software packages. In addition, some packages do not take advantage of the graphics, color, and audio features of today's microcomputers. While constant elaborate displays violate the tenets of good curricular design, the total lack of them makes the microcomputer nothing more than a very expensive teletype terminal.

With the data handling capabilities of a microcomputer, at the very least the program should have some type of lesson management built in to record the student's progress and to assist the teacher in planning for the students individual progress. Ideally, the information concerning student progress would be presented in a manner that would allow it to be used by a number of other programs, in effect "knitting" it into the total computer curriculum.

HUMAN INTERFACING

Students have an almost universal desire to "bomb" the program. They will give unexpected answers, try to change passwords, and generally wreck havoc with the program. A good piece of software will anticipate these attempts, and will prevent the student from doing any damage to the program.

The program should meet several guidelines of good design. The questions should always be asked at the same location on the screen, and in the same format. The program should be consistent in its use of the return key, directions should be clear and straight forward, and important information should be prevented from "scrolling" off the top of the screen.
In addition to the above considerations, there needs to be careful attention paid to the teachers materials that come with the software. A great piece of software will never be used if the teacher doesn't know what it does or how to use it.
At a minimum, there should be behavioral objectives for student progress through use of the program, background information for the teacher concerning the subject matter of the program, and complete directions for the use of the program. In addition, there should be suggested daily lesson plans, pre- and post- tests, student work sheets where appropriate, and cross references to text books covering the subject matter. Nothing should be left up to the teacher to discover about the use of the program.

Often there will be an excellent program that has no appropriate support materials. It is possible to prepare the necessary support materials and make a good program into a great classroom learning package. An example of this would be a good text editor that was not intended to be a learning experience. With the appropriate support materials, it could become the heart of a course on word processing.

These considerations, taken together, make for sound educational software. As a basis for evaluation, or for the design, of software packages, each of the above guidelines must be carefully addressed.

No discussion involving the possible creation of software would be complete without dispelling the myth that anyone can create, easily and quickly, high quality software. It takes time. It requires long and careful planning. It is necessary to field test and modify. In summary, the creation of educational software that is meaningful and effective is not a project that should be lightly undertaken, especially by an educator with already high demands on his/her time.

The most productive approach to the acquisition of software is the carefully undertaken evaluation of commercially available packages. This procedure should include the use of the software in actual classroom settings, an analysis by an experienced computer educator, and a comparison to other similiar packages. A review form, such as the one developed by the National Council of Teachers of Mathematics, is a great benefit in this evaluation.

A computer is only as good as its software. An educator considering educational applications should look first at software. There is much available, but only a small fraction of the total is worthwhile. It is the responsibility of the computer educator to select those packages that are most worthwhile. With this one skill alone, an educator can become a well respected "expert".

- Fred Bockmann

INTERACTIVE LANGUAGE EXPERIENCES BETWEEN THE COMPUTER AND THE STUDENT

Gloria Kuchinskas

The metaphor of the computer as a pencil is one Seymour Papert uses to describe his view of an electronic educational future (Papert, 1981). The pencil is universal; it never dictates what we do with it; it enhances our ability to do anything we want to do with it. When students are allowed to treat the computer as if it were a pencil, then the computer and the student will react with each other. For microcomputers to become a real tool in the langage arts' classroom, teachers must accept this metaphor.

In discussing activities in artificial intelligence - the attempt to make computers smarter - Krutch (1981) uses the term "natural language processing" to describe the same approach to interactive language activities between the student and the computer.

A third proponent of the use of computers for interactive language experiences is Dr. Robert Shostak, a monthly columnist in THE COMUTING TEACHER magazine. His column, "Computers and Teaching English: Bits 'N' Pieces", emphasizes the computer as a language moivator or generator.

Four basic educational principles should be kept in mind when we seek to develop interactive language programs for the computer. One is that in talking about language we are talking about production not reception. In reading we can be purely in the receptive mode. I go from print to meaning when I read. In language production I must either speak or write. For computer language programs the student production mode is writing.

The second educational principle is that childen learn language by modeling. They listen to adults speak and learn to speak. In many cases they listen to adults read and learn to read. They learn language naturally by modeling the language they hear and see. Ineractive language programs should provide the student with some form of language model.

The third educational principle is that the computer itself is a motivational factor. Most teachers will readily accept this point of view. The proliferation of home computing and computer gaming reinforces this point.

The fourth educational principle is that the student should not be a passive receiver of a computer program. As Dewey said a long time ago, children learn by doing.

In addition to these educational principles, this type of computer program requires certain guidelines to be met. The computer should recognize the student; that is, call the student by name whenever possible.

```
10 INPUT "What  is your name ?"
   ;N$
```

From then on the computer uses this name to interact with the stuent.

```
20  PRINT "Help me write a good
    sentence" ;N$
```

This statement by the computer also immediately enlists the support of the student in a joint activity. The child is involved in helpng the computer perform the task.

A second provision should allow the computer to learn from the student. In a sentence expansion exercise, the computer starts with a kernel sentence and asks the student to expand the sentence.

```
30  PRINT "Here is my sentnece
    -----> The man went."
```

```
40  INPUT "Give me three words
    that describe the man."
    ;A$,B$,C$
```

When the student responds with three adjectives, such as, tall, dark and handsome, the computer learns these three adjectives and responds with an expanded sentence.

```
50  PRINT "The ";A$;",";B$;
    ", ";C$;" man went."
    The tall, dark, handsome
    man went.
```

In order to keep a high level of interaction the computer should respond positively to student efforts, regardless of the actual qualitative value of the student response. We know that negative feedback inhibits language production.

```
60  PRINT "Your sentence is
   more interesting than mine."
```

Students should also have a chance to review their own language production and to rewrite if necessary. These revisions could be concerned with spelling, punctuation, usage or syntax.

```
70  INPUT "Look at this
    sentence. Do you want to
    make any changes in
spelling or punctuation
or change any of the words
(Y/N)";R$
```

Student's language production should also be reproduced in some permanent form. This allows the teacher to review it at a later time and allows the student to revise or expand on this language segment in the future. Two reproductive modes are available. If the computer has a printer attached, the computer can be instructed to reproduce the language segment. Since most classrooms will not have printing capacity, the student can be instructed to copy the language segment.

```
80  PRINT "Copy your sentence
    on your paper and show it
    to your teacher later."
```

Interactive programs should also be adaptable to change for coninued use in the classroom without writing a completely new proram. For example, the sentence writing program could be modified by starting with a different kernel sentence.

```
90  PRINT "My sentence is
    -----> People like."
```

The program can also ask the student different questions to expand the kernel sentence.

```
100  INPUT "How many people are
    they talking about";R$
```

The results of such programs should be helpful in expanding stuents' language horizons. If all the students in the class comleted the sentence expansion program and recorded their sentences, all sentences should be different. A class record of these sentenes, with a group discussion, should broaden student vocabularies, concepts and language capabilities. Completing such programs should help students understand how language is generated in written form. After such interactive programs students should accept the fact they can be writers!

Papert, S. MINDSTORMS : CHILDREN, COMPUTERS AND POWERFUL IDEAS. New York: Basic Books, Inc., 1980.

Krutch, J. EXPERIMENTS IN ARTIFICIAL INTELLIGENCE FOR SMALL COMPUTERS. Indianapolis: Howard W. Sams & co.,Inc.,1981

-------, THE COMPUTING TEACHER. Dept. of Computer & Information Science, University of Oregon.

Designing Inservice Computer Literacy Training for Teachers

C. Dianne Martin and Rachelle S. Heller
University of Maryland
College Park, Maryland

Introduction:

Ten years ago the pioneers in the field of computer assisted learning predicted that by 1980 all children would have access to computers in the classroom. Budgetary constraints have prevented most school systems from obtaining large-scale computer equipment needed to reach this goal. However, the advent of low-cost, low-energy microcomputer systems is now allowing many schools to obtain computing power. For the past five years leaders in the field of computer education have been calling for universal computer literacy as the major goal for the instructional use of computers in the 1980s.

In the past, having an awareness and appreciation for computers was sufficient to be computer literate. Now a functional knowledge of computing has been added to the definition. It has been said that computer literacy is "whatever a person needs to know and do with computers in order to function in our information-based society (1)." In the context of American society the classroom is the primary place where computer literacy education will most likely take place.

Through no fault of their own, most teachers at the elementary and secondary levels have not been prepared to teach about computers or use a computer in the teaching of their own discipline. They are "BC" teachers, teachers who received their education Before Computers. A recent Association for Computing Machinery (ACM) report provided recommentations of guidelines for training teachers in computer competencies (2).

Commitment:

There is a growing concensus about the level of computer literacy skills needed by the classroom teacher. The questions that remain deal with commitment, context and content of teacher training programs. Whether the sponsoring agent for inservice training will be at the state, county or local level, the commitment of administrative support and staff resources, not the least of which is financial, is crucial for success.

Currently from state to state there is a great disparity of commitment. A recent survey in the Edcational Learning magazine (3) identified the state of Minnesota as the leader in educational computing. One of the reasons for this is the commitment at the state level to provide inservice teacher training. Alaska, California, Delaware, Florida, North Carolina, Pennsylvania and Texas have all developed state policies concerning educational computing. Interest in the issue is growing throughout all the states. The number one concern expressed by the state educators about educational computing focused on teacher training.

Since there is little hope of federal support for educational computing and teacher training at this time, the responsibility for funding will fall primarily at the state and local levels. In many cases the impetus for funding has come from the grass roots -- PTA'S and local civic clubs. A major concern at this time is providing unified direction for local efforts, especially in the area of teacher training.

School systems are going to have to be committed to spending the money it will take to provide teachers with the necessary skills in computer literacy. Due to shrinking school enrollments, the educational establishment is not being infused with new teachers bringing in new skills and technology, as was true in the past. Rather, the teacher population is comprised primarily of teachers who have been in the system prior to the introduction of computer technology and who are relying on their school sytems to assist them in acquiring these new skills.

Administrators of school systems can demonstrate a commitment to computer literacy by offering professional days for inservice computer literacy training, by creating resource centers and by providing support staff especially in the areas of software and hardware. Administrators of a particular school show commitment to the issues by providing secure storage of computer hardware, by devoting part of teacher meetings to computer literacy, by designating a resource teacher within the school and by being generally supportive toward the teachers' efforts in this area.

Context:

The contextual setting will help determine the content of inservice training. Such factors as attitudes of teachers, availability of computer hardware, numbers of teachers to be trained, amount of time that can be dedicated to training, and the location of the training have a role in determining the content of the course.

Successful inservice training must meet the needs of the classroom teacher. Teachers' attitudes are effected by their classroom environment. The teacher who knows that the newly acquired skills are relevant and useful in the classroom will have a vested interest in the training. A teacher who sees the training as offering "empty skills" with no chance of incorporating this knowledge into the classroom typically reacts negatively to such training.

The availability of hardware for the inservice training as well as for classroom use after the training has a definite impact on the content of the course. If there are sufficient numbers of computers, the course can offer hands-on experience to the participants; otherwise, the course has to be taught in a demonstration mode. The absence of computers does not preclude a training program, but it does limit the scope to include only the awareness level of computer literacy.

The larger the group of teachers to be trained at one time, the less interaction between participants. Training a large group might be done by video presentations or by lectures. Smaller groups are well handled in a workshop format. A tutor, or one-on-one training, is especially relevant when actually using computer equipment.

The time commitment involved can be as great as the time needed for a full three credit college level course to as little as the time needed for an overview of the topic. The commitment of time plays an important role in choosing the type of training to offer.

Another important contextual consideration is the location of the training. Should it be offered at a central location or should it be taken from school to school? The location and portability of equipment is a determining factor in this issue.

Content:

Three credit course:

A 45 hour course in computer literacy which offers 3 college credits or continuing education units can be structured to cover both the awareness level and the functional level of educational computing. Given the necessary equipment, such a course can provide both indepth content material and hands-on computing experience to the teacher. This model can provide concentrated instruction and laboratory experience on such topics as computer history, problem solving, programming, applications and societal impact. There is sufficient time in such a course for the participants to form small working groups in order to develop their own materials. In addition to the lecture content, half of the time in each three hour session can be spent in a hands-on laboratory operating a computer, learning a programming language, writing programs and developing and evaluating courseware (4).

One credit course:

This 15 hour course is a user-oriented course. The content that can be covered is history of computing, computer mathematics, integration of computing skills into the curriculum and social implications of computing. A fifteen hour course does not allow participants to create their own material during the course but it should provide resources for such material development after the course is over. The laboratory component of this course would allow the participants to learn how to use the computer. In addition the laboratory should allow the participants to view, review and evaluate available software. The obvious difference between the one credit and the three credit model is the programming component.

Two-day workshop:

This short course requires that a choice be made between content and hands-on experience environments. A content type course would present, in a lecture style, the following topics: history of computers and numbers systems, social implications of computers and integration of computing into the curriculum. Clearly these issues can only be addressed in an overview or introductory fashion. Exposure to computers would have to be done by demonstrations. Frequently vendors are willing to make hardware and software available for such sessions.

An alternate emphasis for a two-day workshop is a user oriented environment. Such sessions as computers in mathematics, computers in language arts, or planning, purchasing and managing a computer laboraory are appropriate for small group discussions. This workshop could also allow time for communication among the participants concerning their current and future endeavors in computer literacy.

A drawback of this model is that one has to make a choice between content and computer user orientations. When the scope of computer literacy workshops is limited by these choices, the possibility of mismatched expectations of the participants becomes a problem. In addition there is usually not sufficient time to provide meaningful hands-on experience for the participants. Finally, a content oriented model usually precludes time for interaction among participants.

One Day Overview:

There are several options available for the one-day workshop. A first option is to present an overview of computer literacy by addressing such issues as the impact of computers in education and society and the integration of computers into the curriculum and perhaps a small group brainstorming session on how this is relevant to the teacher. A second option is to study one of the previously mentioned topics in depth. An example might be a day long workshop on problem-solving since problem-solving is a skill which needs to be integrated into every discipline. A third options is a hands-on laboratory. In this model a participant would have the opportunity to learn to use a computer, but not to program it.

Follow-up:

Training models should be developed with a built-in mechanism for follow-up and feedback. An important part of follow-up is the commitment to provide opportunities for the teacher to use the newly acquired skills. As with most new skills, one must "use it or lose it." Follow-up should also include the forum for continued communication. This forum might take the form of newsletters, user groups or central software libraries.

A final issue, which may turn out to be the most difficult issue, is that of providing incentives and rewards to teachers in order to keep them in the education field once they have acquired computer literacy skills. Allowing teachers to become upwardly mobile within the system will keep them from moving out of the system to market these skills elsewhere. Several solutions have been proposed. One of them is to use teachers as facilitators to train other teachers in computer literacy skills. Computer literate teachers can also be promoted to resource or curriculum specialists in order to help determine and implement educational computing in their district.

References:

1. Hunter, Beverly. Computer Literacy in Grades K-8. Human Resources Reasearch Organization (HumRRO) presented at ADCIS, Atlanta Georgia, March 1981.
2. Taylor, R.P., Poirot, J.L. and Powell, J.D. Computing Competencies for School Teachers. Proceedings of the National Educational Computing Conference, June 1980.
3. First Annual Survey on State Governments & the New Technology. Electronic Learning, November/December 1981, Vol. 1, No. 2, pages 56-59
4. Martin, C.D., Heller, R.S., Austing, R.H. Computer Literacy: Structuring a Workshop for Elementary School Teachers. Proceedings of the Western Educational Computing Conference, 1981.
5. Heller, R.S., Martin, C.D. Bits 'n Bytes About Computing: A Computer Literacy Primer. Computer Science Press, 1982.

Improving the Validity of Computerized Multiple-Choice Tests in Mathematics

Dr. Hugh McHenry
Department of Mathematical Sciences
Memphis State University
Memphis, Tennessee 38152

A solve and report test is a special type of multiple-choice test in mathematics. Each test question consists of two parts. One is called the problem segment and the other the report segment. Data and instructor observations from a pilot project indicate that the solve and report test is superior to conventional multiple-choice tests in promoting student effort in problem solving, ease of construction, and fairness in grading.

Multiple-choice tests have not been widely accepted for daily use in mathematics. They have generally been considered a poor substitute for traditional tests on which students write-up solutions to problems. Nevertheless, computerized testing and grading, frequently to help with large classes, have led to extensive use of them. This conflict between judgement and practice has created an undercurrent of feeling that computerized testing and grading (i.e. multiple-choice testing) should be used only if the teaching situation makes it impractical to do otherwise.

PROBLEM

Mathematicians argue that multiple-choice tests are invalid for two reasons. In the first place, they squelch creativity. The information given in the distracters provides hints for solving problems that students should develop. Furthermore, the presence of the distracters causes some students to work to get one of the given answers as opposed to concentrating on solving the problem. The worst case of suppressing creativity is when the student resorts to guessing since he is not doing mathematics.

In the second place, the usual ways of grading multiple-choice tests are unfair to students. Minor errors in what are otherwise correct solutions to problems cannot be detected. This means that credit cannot be given for basically correct work. It is felt that partial credit is necessary in order to evaluate students fairly and accurately.

Correcting for guessing is also considered to be unfair. Consider, for example, a four item multiple-choice test with two items designed to test a student's ability to solve quadratic equations and two questions for testing understanding of logarithms. Each question has five distracters. Furthermore, assume that a student knows the material on quadratics but not on logarithms. His grade should be 50%. However, if the correction for guessing whereby 4 points are awarded for a correct answer and -1 for incorrect is used, his grade is given by $(4R-W)100/4(R+W) = (4x2-2)100/16$

or 37.5%. This is lower than it should be since the correction process reduces his credit for knowledge of quadratic equations because he does not know about logarithms. This is not the way teachers grade traditional tests. Poor performance on one problem does not lead to loss of credit on one gotten correct.

If no correcting is done, lucky guessing makes the grade incorrect and the test invalid. These problems are inherent in correcting for guessing and are felt to be serious weaknesses in multiple-choice tests in mathematics.

SOLVE AND REPORT TESTS

Because of the many advantages in using a computer with multiple-choice type tests as well as those inherent in such tests themselves, a project was initiated to develop a multiple-choice type test for mathematics that would not be subject to the criticisms listed above. The remainder of this report describes the type of test developed and gives data from a pilot project.

The test is called a Solve and Report (S&R) test. Figure 1 gives an example of an S&R test item. It consist of two parts called the problem segment and the report segment. The problem segment contains a statement of the problem to be solved. The student is to write-up a solution to the problem as he would on a conventional test. After he gets results that he considers to be the solution to the problem, he proceeds to the report segment. This segment consists of five true-false questions about the solution. The student identifies all the statements which are true and marks the corresponding identifying letters on his answer sheet. The questions are designed to explore major aspects of the solution.

Grading an S&R test can be done in a variety of ways. The one used in this report was selected because it appears to be a reasonable compromise on correcting for guessing and giving partial credit. Each problem is graded in isolation as though it were the only problem on the test. The total test grade is the sum of the respective problem grades.

Let R be the number of the five statements correctly labeled as true or false and W the number of incorrect classifications for a given problem. Then $T = (R - W)/5$ is calculated as the raw grade on the problem. If $W < .6$, then W is set equal to zero. Let P be the number of possible points allocated to the problem by the teacher, then the grade on the problem is defined to be TxP. A 10 point question where a student misses one of

FIGURE 1. Example of a Solve and Report Test Item

---

1. Sketch the graph for $y = 2x^2 - x - 6$.

   A. The vertex is in Quadrant IV.
   B. If (x_1, y_1) and (x_2, y_2) are the x-intercepts, then $x_1 + x_2 = 1/2$.
   C. The graph is a parabola.
   D. The graph lies entirely on one side of the x-axis.
   E. Let (x,y) be the vertex and (u,v) the y-intercept. Then $xyv + u = 39/4$.

---

the five questions would yield R=4, W=1, P=10 for a grade of (4-1)10/5 = 6. The grade on an entire test is the sum of the problem grades converted to the percent of the total points possible.

RESULTS

A pilot project was conducted in which an S&R test was compared to a traditional multiple-choice test. Two forms of the final examination in a college elementary algebra course (high school algebra I level) were prepared. Each form used exactly the same 31 problems. The traditional test (control group) used five answer choices on each item with exactly one being correct. The S&R test (experimental group) was prepared according to the example in Figure 1. Both of them were prepared by the same person. Each of the 31 problems was weighted equally on each test. Grades for the control group were simply the percent of answers that were correct. No correction for guessing was used. The experimental test was graded using the method outlined above.

A total of 114 students in 8 classes took the tests. There were 61 in the control group and 53 in the experimental group. The classes were taught by 6 instructors. Equal numbers of students in each class took the tests in order to balance out the impact of teacher and class differences. The two groups did not end up with equal numbers because of odd numbers of students in classes and instructors not adequately controlling the distribution of tests to students arriving late.

The two treatment groups were compared on prefinal class averages as a check on comparability. Seven standard type multiple-choice tests were given during the semester. Each student's average on these was used as his prefinal course grade. Table 1 gives the means and standard deviations for the two groups. The t and F statistics at the bottom of the table are not significant at the 5% level. Therefore, the two groups were accepted as comparable in terms of average and variability. The N = 96 in the table is not equal to the 114 who took the final tests because one graduate student instructor lost part of his class records.

TABLE 1. Comparison of Experimental and Control Groups on Prefinal Course Grades

GROUP	N	MEAN	S.D.
Experimental	45	61.4	13.7
Control	51	63.7	15.3
t = .77		F = 1.23	

Table 2 gives the basic statistics for the two types of final exams. The statistics indicate that the two test are very similar with respect to range of grades, variability, and Kuder-Richardson test reliability.

The major difference is that the average on the S&R test is 21.4 percentage points below the traditional test. Since there were five choices on each question on the traditional test and no correction factor was used, the expected profit from guessing was 20 percentage points. Therefore, the difference in the two averages indicates that the correcting for guessing on the S&R test worked as planned.

Further comparison of the two types of tests was made by determining how well grades on each

TABLE 2. Basic Statistics for the Solve and Report and Traditional Multiple-Choice Tests

Statistics	Test Type	
	Traditional	S & R
N	61	53
Mean	61.6	40.2
Median	61.3	38.1
High Score	90.3	87.7
Low Score	16.1	11.0
Standard Dev.	19.2	15.5
Reliability	.86	.88

correlated with the prefinal course averages. For the traditional test r = .84 (N=51) was found. The value for the S&R test was r = .85 (N=45). It is evident that the two types of tests correlate equally well with other course grades.

Difficulty measures for each item on the two tests were computed. It has already been observed from Table 2 that the S&R test is the more difficult of the two. A matched pairs t-test where the test questions are the subjects yielded t = 5.25 with s.d. = 22.7 and d.f. = 30. This is a significant difference since t(.05,30) = 2.04. The difference appears to be due to correcting for guessing on the S&R test but not on the other. Therefore, the issue with difficulty is whether the difficult questions on the S&R test are also the difficult ones on the traditional test. The correlation between the two sets of difficulty measures for the 31 questions was calculated and produced r = .73 where r(.05,30)=.31). It appears that relative item difficulty is consistent across examinations.

Item discrimination was also analyzed. The mean discrimination value for the S&R test was 33.5 with s.d. = 14.7. For the traditional multiple-choice test the mean was 42.3 with s.d. = 17.4. N = 31 in both cases. A matched pairs t-test gave t = 3.02 with d.f. = 30 where t(.05,30) = 2.04. Thus, there is evidence that the two tests do not discriminate equally well between students who do well and those who don't. Furthermore, the correlation between the two sets of measures was r = .21. These statistics indicate that a question which has good discrimination on one of the tests will not necessarily be the same way on the other test.

Two additional points emerged from the pilot project. 1. Students worked harder and did more mathematics on the S&R test. The instructors unanimously reported that the S&R students spent longer on the test and were more creative in solving problems than the others. 2. S&R tests are easier to construct. Fewer questions are needed since each problem involves more mathematical details and students take longer to do the work. It is also easier to construct the five questions for a problem than it is to find five good distracters on a traditional problem.

CONCLUSIONS

The conclusion is that solve and report type multiple-choice tests are superior to traditional multiple-choice tests in mathematics. They require students to do more mathematics and encourage them to be creative in finding solutions. They are easier to construct and the statistcs from the pilot project indicate that on other points they are reasonable substitutes for traditional tests. Therefore, they should make objective testing more acceptable to the mathematics community and thereby promote the us of computers in testing and grading.

INSTRUCTIONAL USES OF COMPUTERS: THE GOOD, THE BAD, AND THE UGLY

Sally A. Sloan
Minneapolis Public Schools

There is, for the most part, a consensus of opinion about instructional uses of computers. Regardless whether the delivery system is a large sophisticated timeshare computer or inexpensive microcomputers, most educators involved in instructional uses agree that computers are excellent tools to:

Introduce Concepts,
Reinforce Concepts,
Practice Skills, and
Create Learning Environments or Settings.

To help accomplish these tasks a large body of packaged educational software now exists in formats generally referred to as:

Demonstration Programs,
Simulations,
Drill and Practice,
Learning Games, and
Tutorials.

In addition, many educators subscribe to the notion that logical thinking and problem solving skills are enhanced by learning to program the computer, then using the computer to explore and solve problems.

Some controversies appear when educators begin to ask:

"who" should use which sorts of programs,
"when" in the regular curriculum do computer applications belong,
"what" is the role of the computer in particular subjects, and
"where" in the school should computers (and/or terminals) be placed?

These are some of the questions which occur in the instructional environment: the classroom, media center, study hall, computer room. Other questions arise in the area commonly called administrative uses, those areas related to but not directly involved with instruction such as classroom management, attendance scheduling, office needs, district needs, etc.? The controversies seem to grow as the number of computers in homes increases and the quantity of commercial software, both instructional and recreational, mushrooms.

It appears that the issues are the same whatever the environment in which computers might be used.

1) What are appropriate uses?
2) What are ethical uses?
3) What are legal uses?
4) And what can we do about any of it?

Let us look at the basic objectives of using computers in education more closely. The notion of "Introducing Concepts" via a computer is not new. In early developments such as the Huntington simulations, concepts in Social Science, Physical Science and Economics were often intruduced in the sense of allowing students to "discover" them while participating in the simulation. That notion of "discovery learning" has been accepted by the education philosophers for some time. The ready availability of computers has only recently made it practical. That students learn concepts they have "discovered" more thoroughly and retain the knowledge longer has been documented by educational research. In recent months, computer programs which "demonstrate" certain concepts have begun to appear. The demand for this type of program, one by which a teacher with some ease could convey a complex concept, is strong. That leads us right back to the question of appropriateness. Not all concepts lend themselves to demonstration on a screen or paper. But even for those that do, such as the slope of a line based on its equation, there still remains the questions who, when, ..., etc. Taking just that one concept, the slope of a line based on the equation for that line, some of the answers may seem obvious, yet are often contested. This concept is normally taught in regular eighth grade mathematics classes. As a demonstration format program it works well handled by the teacher, using a single computer, in front of the whole class. Yet it could also be used by one or more students to "explore" the notion and "discover" on their own. Since the program allows the user to input any number of equations and observe the line on a graph, it has been called (and used!) by some as a drill. Before the computer program, a common technique used in teaching this concept was to have the students draw a great many graphs of lines given their equations. Rather than change that style of teaching the concept, some teachers simply insert the computer where graph paper and pencil used to fit. Most of us would call that an inappropriate use of the computer. A demonstration program such as SLOPE, used to INTRODUCE a concept well known to occur at a particular place in the curriculum may not generate controversy regarding "when" it should be used but is still open to the other questions. In a school where the computers have been designated "for remediation" what will be the chances of a an eighth grade math teacher getting use of the machine to demonstrate a concept to a regular class? Likewise in situations where the computers are believed to be "best" used in classes for the gifted. A recent research project by an eastern college referred to cities where decisions had been made at top administrative level to limit use of computers to particular segments of the population. In one where they were used entirely for remediation the regular and gifted students felt some stigma attached to computer uses, an attitude likely to close many doors to them in their futures in our highly technological world. The other side of the coin, in the city where only the academically able were allowed to use computers in the elementary school, staff of secondary schools found that underachievers, who might have benefited greatly by computer assisted learning felt they could not possibly handle them and were reluctant to try. When we ask "what" the role of the computer might be in a particular subject, we are again asking "what is appropriate." Perhaps at some point in the curriculum of every subject there is a place for the computer where it will accomplish a teaching task better than any other means. That is NOT to say that at every point in every curriculum the computer is the answer to teaching the subject. There is a time to review the rules of a sport using a computer, perhaps even to simulate playing the sport as a way to develop strategies for play, but there is also a time to put on your sneakers and get out on the field. It is the responsibility of educators to learn how to discriminate, how to recognize "appropriate" uses of computers. Even the "where" becomes a bone of contention in many schools. Should the computer be in the media center with students sent out of class to use certain programs? Should it be in the classroom, like an overhead, for the use of the teacher and the class at that important teachable moment or should it reside in some special "computer room" staffed by parent volunteers or school aides and used freely by any student who so desires? What is "appropriate"? Perhaps the best answer to the above set of questions is YES! ALL of the above! When it is educationally appropriate! A funded project in a small Minnesota town adopted a slogan that is worth repeating: "When you are considering the role of computers in education you should not be thinking about computers, you should be thinking about education!"

There are rarely simple answers to complex questions. The case of appropriate uses of computers in instruction is a complex question. One solution is training. Training staff, ALL staff, in computer literacy: what computers can and cannot do, the impact on society, hardware operation and sufficient programming to really believe that people control computers, is only the beginning. Staff must have training on teaching methodology and strategies using the computer as well as opportunities to review the available software. A trained staff can make better decisions about when, where, what and who should be using the computer. Best of all a trained staff can incorporate the computer into the regular curriculum in a way which will make it a part of the students' school experience in a natural yet thorough way. Our responsibility as educators to graduate computer literate students is not in question. Yet this apparently simple solution to the question of appropriateness of uses is indeed complex. HOW can a district accomplish that much training? HOW can a district acquire sufficient machines to complement the curriculum? For that matter, HOW MANY computers does it take to implement instructional uses? Should they be the same units used for administrative uses? No, there are not simple solutions to the problem. What we do have is a wealth of experience from all over the country, spanning more than a decade of successful uses of computers in education. Meetings such as this, the AEDS National Conference, are one method of disseminating the results of that experience. Here is where we learn how one district "subverted" the schools into buying computers by loaning them one until they were deeply involved, then retracting the loan: result - school buys computer. Here is where we learn how a district motivates teachers to demand training by supporting just one lively active teacher in a school. When the rest of the staff observes the joy and successes connected to computer use in teaching, the demand reaches proportions well calculated to respond to offers of in-service training during free hours. Here is where we learn the training techniques to use with administrators as opposed to groups of elementary teachers and the things to watch for when training secondary teachers and media specialists. We learn from each other. We also learn by sad trial and error. Some of the other questions asked in the introduction are fairly new. There is no body of experienced districts to approach seeking answers. We will have to discuss the problems, suggest solutions and wait. Hopefully, it will not be long before results begin to show.

Two of the issues raised at the beginning of this paper are relatively new or at least have just recently begun to be realized as serious issues. They are the questions of ethics and legality.

No one ever denies that copyright laws exist. At most, someone may say that they are unclear or should be changed. Nevertheless, copyright laws exist and apply to computer programs. There is an old story among pilots of retractable gear airplanes to the effect that it is not a question of whether or not you will one day make a gear-up landing but a matter of WHEN you make your gear-up landing. The same may be said, these days, of WHEN you make your first illegal copy of copyrighted software.

People in positions like mine are often asked to serve on committees discussing the impact of technology on the curriculum, e.g., "what changes will be caused by the very presence of the technology?" We might also be asked: what changes in values in our society are being caused by the presence and nature of computer technology? The popular media, reporting economic research, tells us that information is becoming the major product of our country. We are besieged by organizations concerned with "privacy" as an issue in our government. States are passing "privacy acts." The general public is legitimately concerned with the uses to which the immense amount of easily accumulated data on each of us may be put. What "ethical" uses exist for the easily accumulated data on students being stored on computer memories all over the country? It is good teaching practice to group students by the learning objectives on which they need help in order to provide timely intervention. Classroom management programs which analyze test results, report student achievement based on teacher (or school, or District) determined objectives and offer suggested prescriptions are an extremely popular type of computer program. The current technology allows that classroom level information to be transmitted to larger memory banks wherein the entire student records are maintained including attendance, health, socio-economic information, etc. I have never heard of such records being used unethically but potential is certainly there. What a bonanza for commercial sales representatives from everything from "auxiliary schools" (Is your child having trouble in school? Send him to us...for only $$$ we will guarantee...) to computer companies (Buy a brand X computer and let your child learn at home). For years educational institutions have resisted the pressure of the commercial domain for mailing lists and personal information. How do we insure, in a technological society, that personal information is not easily stolen? Perhaps more difficult to answer but closer to home are the questions of how we insure that the great masses of information about students and staff are used ethically. Is it ethical to perform a computer survey of the achievement of each year's students, over several years, to determine the effectiveness and continued employment of a particular teacher? Is it ethical for a teacher to review a student's past history before beginning to teach that student? How much information needs to be kept on anyone? Who should have access to that information? Just because information is so easy and inexpensive to store must we really store everything?

The question of ethical uses is not limited to the storage of information or privacy issues. There is no "law" covering one student (or teacher) erasing the computer programs stored by someone else on an instructional computer. It is not patently illegal. Whether or not it is unethical, whose responsibility it is to decide that question, and whose responsibility it is to "teach" the decision, seems to depend on which end of the situation you reside. If you are in the middle of a long, complex, to you important, program and run out of space, you may decide that destroying someone else's program to get some space is justifiable. If you are the student whose program was destroyed you may not agree with the decision which was made. If you are a teacher who has occasionally zapped someone else's program yourself you may not be willing to take on the task of teaching others that it is wrong.

The student who, in a zealous pursuit of information about how computers function, crashes the instructional timeshare system probably does not feel the experience was either unethical or illegal. Was it either? That is not a trivial question these days. We find ourselves "tempering" judgment based on the circumstances. Should the student who inadvertenly crashes a computer and immediately reports the fact and how it was done be treated exactly the same as the student who deliberately sets out to crash the system? Are we dealing with accidental homicide versus premeditated murder? Who decides? Should one be labeled "bad" and the other merely "ugly"? We are seeing many of society's deepest problems repeated in a different environment. Before laughing off the crash of a computer system as far different from murder consider that the students of today have access to terminals, in homes, libraries, schools, etc., and seem always to have "IDs" to numerous computers including several which have the potential for serious effect on the rest of the world. It is one thing the "crash" a system dedicated to K-12 instruction. It is another to crash the phone company, American Express or the Defense computers. Yet the process and many of the techniques are the same. Yes, theft may be involved. If a student "steals" the "ID" to a State department computer or is given it by a member of the family who works there and thinks it OK for junior to use the machine at night - the result is the same after the crash. Is anyone liable? Can we stop it? Students with whom I work have only one suggestion. They claim there is no way to stop them from getting access to various computers, that we can only teach them that it is wrong! But can we do that? When a teacher is faced with a room full of eager programmers and the instructional computer fails and one of the students says, "my Dad works for the phone company and I have an access code," what do you suppose that teacher is going to do? It is a very difficult decision. It sounds like the days of riots and reform to speak of providing sufficient computer power, supplies and software so that teachers and students are not motivated to break copyrights or steal codes or destroy each other's programs. Has our societal value system changed to the point where a personal perceived need has precedence over private property? Some companies prosecute copyright infringements of software. Are staff ever fired for copying a copyrighted book or diskette? Programmers are warned of the impossibility of protecting their products. Are we seeing tacit acceptance of lawbreaking depending on the circumstances? It has been said that when a law is known more for the breaking than the keeping (like prohibition) it is a bad law and should be changed. Are we approaching a time when copyright laws are revoked because they cannot be enforced because no one (but authors) wants them?

In summary, we face an exciting time when technology has presented us with a tool that diminished the drudgework associated with education, provides better management of learning, saves time useful in instruction and offers a fantastic potential for improved learning. With all those very "good" uses of computers in education comes the potential for "ugly" misuses of good instructional materials, addictions to arcade games, unethical behavior toward others' programs and blurring of values in our society. The "bad" uses of computers, computer crimes, control of society through the use of sensitive information, etc., have added an additional dimension of fear to a world already terrified by the proliferation of nuclear armament. It is, perhaps, a part of our responsibility as educators to show the world, especially the young, that the advantages and benefits of computer technology, used wisely and well, far outweigh the hazards. A more computer literate society can guard itself from many bad and ugly uses, e.g., the loss of millions of dollars to crime and error in banking could have been avoided had bankers understood the round-off/truncation problems of computer designs. The students reassure us that they at least do not fear misuse of information they feel confidant they could erase at will - an interesting twist to the question of ethics. That becoming technologically knowledgeable is difficult cannot be denied; that it is imperative in our society must be acknowledged.

TEACHING YOUNG CHILDREN TO PROGRAM IN BASIC[1]

by
Royal VanHorn, Associate Professor
Division of Curriculum and Instruction
University of No. Florida, Jacksonville

The following conversation occurred early this year between a second grader and her teacher.

Teacher: "To light a block on the CRT you first count how far over the block is. Then, you count how far down it is. See, (referring to a graphics layout chart) this block is: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12 over and 0, 1, 2, 3, 4, 5, 6, 7, 8 down. So, its address is (12,8)."

Student: "I know, but why do I have to count? Can't I just use these numbers (pointing to the horizontal and vertical axis on the chart)?

Teacher: "Yes you can--that's certainly easier!"

Young children in the primary grades (1-3) can learn to program in BASIC and presently are in hundreds of schools around the country. The primary purpose of this paper is not to consider whether young children should be taught, or allowed to learn, to program. The answer to that question seems obvious. Consider an analogy to the piano. Player pianos used to be quite popular, now they're museum pieces. People, in the long-run, preferred to learn how to play the piano. The same will probably happen with computers. Besides, it's alot cheaper and easier to let kids learn to program than it is to teach them school subjects using a computer. The latter generally requires more sophisticated computers and expensive software.

The purpose of this paper is rather to consider three different approaches to teaching young children and other beginners to program; using programming manuals (written at an appropriate reading level) using computer assisted instruction software, and the direct instruction approach. Each approach and its advantages will be discussed separately. Suggestions for using each approach will also be offered.

Programming Manuals

> "The problem is I lived half my life before I learned it was a do-it-yourself job."[2]

Kids can and do learn a lot on their own. Let them have access to a microcomputer, but tell them they'll have to read the manual to know how to run it, and you've got a group of highly motivated kids. Even reluctant readers will sit down and read the manual if they know it's the only way they're going to be able to get the computer to work. Obviously, this approach requires that you use manuals which have an appropriate reading level. (A series by this author has a measured reading level of second grade. Others are producing similar manuals.)[3]

Using programming manuals to teach programming has many advantages. First, students get experience in reading while they're learning to program, i.e., they have a real reason to read a book.

Second, while using the manuals, kids learn to follow directions and depend upon programming manuals. This is a real world situation--every programmer depends on his or her ability to learn from technical manuals.

Third, the manuals are "machine specific" and therefore BASIC language differences from computer to computer are not a problem. (Not all BASICs are alike.)

The fourth advantage is students can go at their own pace. It's alright if they get ahead or fall behind. And, if they haven't read the manuals for awhile, or they've been absent, some manuals provide "things to remember" sections which allow for quick review.

Fifth, another real advantage is that kids can write in the books or take notes in the margins. They can even highlight important passages with a bright colored marker. This aids retention of important BASIC words.

Sixth, students learn to program with minimal adult supervision. This frees the teacher to work with one group of students while another group is at the computer.

Suggestions for Using Programming Manuals

To achieve the best results using programming manuals:

1. Do an ample number of introductory activities to develop a bit of computer literacy in students.
2. Thoroughly orient students to the idiosyncracies of the particular computer they'll be using, such as how to turn it on, load a disc, etc.
3. Acquire one copy of the manual for each student so they can take notes in the manuals and study them when they're not at the computer.
4. Pair-up students. When two students use a computer at once, one will often see typing errors the other misses, and understand words the other can't comprehend. In addition, students will get more turns at the computer each week if they're paired up. Also, you can pair a good reader with a poor one.
5. Be available to help students who have questions or have an older, more capable student serve as a "peer tutor."
6. If you're teaching very young children (second grade or lower) or if you have a group of poor readers, you may want to provide an adult or older student "reader." Senior citizens and parent volunteers make good readers. (Be sure to tell them to "sit on their hands" and not push keys for the kids!)
7. Use time on the computer as a reward for students who get their schoolwork done correctly. Use computer time only as a reward--never make a kid miss his turn at the computer as punishment.
8. Be ready to suggest other books students might enjoy reading that will help them learn even more about programming.

Computer Assisted Instruction

A second approach to teaching programming involves using Computer Assisted Instruction or C.A.I. In this approach, a series of pre-prepared lessons (computer programs), recorded on computer diskettes or cassettes, are loaded into a computer. Then, students interact with a lesson (program) which is presented by the computer. One series of 14 such lessons, available for use on a variety of microcomputers, is available from Sterling Swift Publishing Company.[4] Lessons in this particular series range in difficulty from "how to use a computer keyboard" to much more advanced topics. A manual or workbook is often included which reinforces and extends the material covered in the C.A.I. lessons.

This method of teaching computer programming also has its advantages. First, kids participate in one major application of computers, that of C.A.I., as they learn to program. Second, the approach is quite motivational--most people enjoy interacting with a computer. Third, and indeed a significant advantage, students get to see various small programs run while they're interacting with the C.A.I.

47

program. By doing this, students can easily conceptualize what a program is and what a program does. Fourth, less reading is involved in this approach which can either be advantageous or disadvantageous, depending upon the setting and the students.

The C.A.I. approach and the programming manuals approach share several advantages. Using the C.A.I. approach allows students to proceed at their own rate, and thus the teacher can work with one group while another group is at the computer(s). And, because the C.A.I. programs are "machine specific" BASIC language dialect problems are avoided.

Suggestions for Using C.A.I. to Teach Programming

1. Do an ample number of introductory activities to develop some "computer literacy" in your students.
2. Thoroughly orient students to loading the lessons and handling the tapes and disks.
3. Acquire enough copies of the accompanying manuals so each student has a copy. Assign the appropriate lessons and discuss them with students.
4. Have students, especially elementary students, practice using the new BASIC words learned in each C.A.I. lesson before progressing to the next lesson. The only real way anyone can learn to program is to actually write a few programs.
5. It is probably best to have students do the first several lessons individually, but students may be paired up for later lessons. (Every student should personally type on the keyboard of the computer during the keyboard lessons.)
6. Encourage students to try to write working programs of their own design.
7. Supplement these C.A.I. lessons by having contests for the neatest program which incorporates the new words learned in each lesson.

It might be possible to combine the first two approaches, i.e., use both the programming manuals and the C.A.I. lessons simultaneously to teach programming.

These two approaches, however, usually introduce topics in radically different orders. This probably wouldn't bother high school kids and adults, but it might confuse younger students. You are encouraged to innovatively try to combine the two methods.

Direct Instruction

Approach three is a more traditional one involving the direct teaching of lessons on programming. These lessons can take many forms but using well laid-out overhead transparencies is a fine approach. At present such transparencies are commercially available from a variety of sources.[5] Teachers who use this approach usually prepare "handouts" containing short example programs for students to enter on the computer. This is a great idea and teachers using any of the three approaches discussed herein are encouraged to give dozens of examples to their students.

The direct teaching approach has one very distinct advantage--the teacher should be able to sense whether or not students understand a lesson by collecting feedback from the class while the lesson is being taught. In this approach students do not generally progress at their own rates, which can be both advantageous and disadvantageous, depending on the setting.

A second major advantage of this approach is that students are free to ask questions which can be immediately answered by the teacher.

A third advantage, which is shared somewhat by the first approach, is that students don't tie up the computer while new words/ideas are introduced. Students learn in class and then apply what they've learned on the computer.

Regardless of the approach, when the teacher senses that a number of students are having trouble with the same aspect of programming, calling the students together for a short lesson is particularly valuable.

Suggestions for the Direct Teaching Approach

1. Make the lessons short and interesting. Fifteen minutes is about right for elementary school students, twenty to thirty minutes for older students. (If students are bored with the lesson they might think programming is boring.)
2. Prepare all handouts, visuals, e.g., transparencies, charts, and the chalkboard, ahead of time and double check them for accuracy.
3. If you plan to show students programs, be certain the programs will actually run before you display them. The only way to be sure is to type them into the computer and run them before you use them as examples.
4. In your lessons, always point out the seemingly insignificant things such as punctuation which can radically affect the output of a program.
5. Give some thought before beginning to the sequence you'll follow to introduce new topics or new BASIC words. Sometimes a seemingly sensible sequence doesn't work. For example, young children often don't find PRINT "   " statements interesting--they'd rather just type on the computer.

There is undoubtedly a time and a place for all three methods of teaching programming. The essence of good teaching is variety. There is no one right way to teach, but teaching using only one method gets pretty monotonous. Better teachers will artfully use a combination of the above methods to teach programming and profit from the above suggestions.

REFERENCES

1. Portions of the paper originally appeared in VanHorn, Royal, Teaching Computer Programming to Beginners. Austin, Texas: Sterling Swift Publishing Co., 1982, 39-47.
2. Samples, Robert, Environmental Studies Project Poster, Boulder, Colorado: American Geological Institute, _____.
3. At least three companies are producing programming manuals for elementary students:
    1. Larson, Sally G., Computers For Kids, Morristown, New Jersey: Creative Computing Press, 1981. (TRS-80, Atari, and Apple editions)
    2. Taitt, Henry A., Creative Programming, Charleston, Illinois, ____. (TRS-80, Apple, PET, Vic 20, and TI99/4 editions available in multiple volume format)
    3. VanHorn, Royal, Computer Programming for Kids and Other Beginners, Austin, Texas: Sterling Swift Publishing Co., 1982. (TRS-80, Apple, and TRS-80 Color computer editions)
4. Poirot, Jim and Retzloff, Don, "How to Program In BASIC," Austin, Texas: Sterling Swift Publishing Co., 1979 and 1980. (Apple, TRS-80 and Commodore PET editions) Note: Other companies produce a variety of aids to help people learn to program their computers.
5. Transparency Masters useful for teaching programming are available from at least two sources:

    _____, Visual Masters for Teaching BASIC Programming,[2] Ormond Beach, Florida: Camelot Publishing Company, 1982.

    VanHorn, Royal, Teaching Computer Programming to Beginners, Austin, Texas: Sterling Swift Publishing Co., 1982. (See the Appendices)

MICROCOMPUTER INSTRUCTIONAL SUPPORT IN THE CLASSROOM (MISIC) PROJECT - MICROCOMPUTERS IN THE CLASSROOM K-12, A 'HOW TO DO IT' REPORT

Paul A. Becht
Sara-Maria F. Becht

University Of Florida
Gainesville, FL 32611

Computers in the classroom have been a reality for sometime, but in very limited numbers and only for specific disciplines such as science and math. Not only did the teacher have to be a programmer but an extensive budget was needed to support the hardware. Both of these factors severly limited this technology in the classroom. That was before the microcomputer became available.

Consider the scenario. You walk into your classroom on the first day of class and the curriculum coordinator meets you in the doorway. You are shown a new microcomputer on your desk and told the superintendent said all teachers are to use the computer in the classroom and for you to prepare your students to be computer literate. You are told "good luck" and left to your class. What are you going to do? Think about it.

Most teachers have little or no idea what a microcomputer is or how to use one. In 1979, this problem came to the attention of a number of school district administrators in North Florida. The Microcomputer Instructional Support In The Classroom (MISIC) Project was initiated to investigate this problem and develop a process for effective implementation of microcomputer hardware and software into classrooms.

Six objectives were developed for the project.

1. Produce a computer-literate student who will be able to cope with and succeed in the future computer-technological world.
2. Prepare teachers to use microcomputers in the classroom as an active integral and functional part of the curriculum.
3. Provide hardware and software support for the classroom teacher.
4. Prepare teachers to evaluate and report on the current state of the art instructional software.
5. Develop and support an instructional computing resource center, the MISIC Center, that will provide support for software evaluations, hardware use and in-service training.
6. Provide a model for implementing microcomputers into classrooms in Florida Public Schools K-12.

Funding for the Project was sought from a number of government agencies, both federal and state.

The MISIC Project was partially funded for software by the Florida Department Of Education through the Northeast Florida Educational Computer Consortium, a consortium consisting of nineteen school districts, two universities and two junior colleges. Hardware support, in the form of eight loaned TRS-80 single disk Model I microcomputers, was obtained from Radio Shack, Tandy Corporation, Fort Worth, TX.

The study involved nine classrooms, three elementary, three middle school, and three high school. Each classroom would have access to a microcomputer. The Project classrooms were spread over three school districts in North

Florida and in the elementary and high school programs at P.K. Yonge Laboratory School.

The Project distributed over one-hundred and fifty different commercial program packages to the Project classrooms with many of the packages containing two to five programs each. Also, each classroom received over thirty blank diskettes for program transfer. Three sets of the one-hundred and fifty programs were delivered because there were three classrooms at each of the three (elementary, middle and high) grade levels giving an average cost of about $36 for a program package after deducting the cost for blank diskettes. Project teachers were asked to evaluate each software package using the software evaluation form found in "The Computing Teacher". These evaluations are being compiled and placed on a data base. Copies will be available in late Summer 1982. In-service programs were provided for the Project teachers and quarterly reports were turned in reporting on the progress of the Project in the classroom. Visitations to the Project classrooms were made by the Project staff and video tapes were made of classroom activities using the microcomputer.

RESULTS:

The MISIC Project was designed to span two school years. At this writing the Project is still on-going and all data has not been collected. However, some important preliminary results have surfaced. They are as follows:

1. Teachers need a computer awareness training program with sufficient hands-on experience.
2. Most teachers do not need or want training in programming or programming languages.
3. Microcomputer hardware and software needs to be chosen carefully to meet specific classroom needs.
4. Student programmer assistants become an integral part of successful use of microcomputers in classroom settings.
5. A single microcomputer can become an effective classroom tool if the teacher has sufficient support.
6. Single microcomputer classrooms have been successful using computers:
    a) as a learning center
    b) with multi-monitors
    c) with TV Projectors
    d) as an electronic blackboard.
7. Ample commercial educational software is available but teacher-done software evaluations are needed for intelligent purchasing of useable software.
8. A microcomputer laboratory is the most effective way to teach computer literacy and programming.

These findings are based on the data currently available. A more complete report will be available in late Summer 1982.

CAI: PRACTICE, ART,
AND SOMETIMES SCIENCE

Robert L. Burke, Ed.D., Director
Computer Education Laboratory,
Nova University, Ft. Lauderdale

The basis for this paper is the proposed creation of a knowledge base to assist the author of computer assisted instruction (CAI) courseware. The most unique feature of the proposed CAI knowledge base would be its structure. Patterned after the artificial intelligence concept of an "expert system", it would be organized in terms of plainly written rules. Each rule would be supported by a variety of evidence, ranging from conventional wisdom rationales to hard research conclusions. It is hoped that the development of such a knowledge base will facilitate the unified development of the practice, art and science of CAI courseware authoring.

This paper is based upon three premises. Currently there is no such thing as a unified knowledge base to serve CAI authors. A CAI knowledge base can be developed. The structure of the CAI knowledge base may be theprimary determinant of its success. The major sources of information for a CAI knowledge base are identified and discussed. An approach which would synthesize a widely diverse set of elements from a variety of sources into a CAI knowledge base that is universally useful, and relatively free of jargon and bias. The paper goes on to sketch the framework and describe the elements of such a knowledge base.

PREMISE I:

There is no unified CAI knowledge base to serve CAI authors.

It is easy to say that there is little direct connection between the CAI author and the research in human learning which might provide a knowledge base for CAI practice. Having said that, it is also easy to condemn either the CAI author for not heeding the teachings of science or the researcher for not communicating research findings to CAI authors effectively. To suggest a workable solution is much more difficult.

While there is no unified, comprehensive knowledge base for CAI authors, there is considerable conventional wisdom among practitioners. Every day, the CAI author must make decisions about CAI practice. Much of the activity of CAI authors can be better characterized as art than science. The CAI author may always have much in common with the poet.

The practice of each CAI author is shaped by a variety of forces. Right now, I'm afraid that the most salient of these are market forces. This is certainly not true in all cases and it can be expected that market forces will decrease in significance as determiners of CAI practice.

The training of the CAI author is another factor which has a great effect upon the type and quality of CAI that a given author produces. An author with formal training in the design of individualized instructional systems will create very different CAI from an author who lacks such training. An individual with extensive formal training in computer science will be able to create much more exotic CAI courseware than someone who does not have such training.

The CAI author's experience is another strong factor in determining the kind of CAI courseware that will be developed. CAI courseware developed by the individual with long experience in the development of programmed learning will almost certainly reflect that experience. The individual experienced in video production may see CAI as a "neat" way to control the presentation of video segments. The textbook author may see CAI as just another "genre" of writing.

Each CAI author tends to emphasize the characteristics of CAI practice that reflect the author's orientation, experience and training. Because of the vast diversity of people now authoring CAI there is no great degree of consistency in practice nor in the courseware produced.

The current great diversity among CAI authors can have a salutory effect on CAI – a rich variety of new models of CAI may be developed which might not otherwise have been created. On the other hand, a lot of "wheels" will get "reinvented", a lot of synergy will be lost, and the science of CAI will be much slower to be developed.

To point to the lack of a science of CAI is not to say that the practice of CAI authoring is not undergoing extensive development. There are a great many more people now wrestling with the development of CAI courseware than there were five years ago, and more getting involved everyday.

This veritable army of budding CAI authors finds little opportunity for formal training in CAI authoring. They find a literature of CAI which is sparse, very diverse, uneven in quality and of very little real help. My new book, CAI Sourcebook is intended to address this need. It draws together, organizes, and relates much of the body of knowledge of CAI authoring to microcomputer CAI courseware development.

PREMISE II:

There is information in many different places and of many different types that could be included in a CAI knowledge base.

There are two main sources which can be used to construct a CAI knowledge base – current practice, and the literature of research on human learning.

Study of current practice in CAI courseware production includes the collection of the considerable conventional wisdom, and is such a rich source that the collection process will be never-ending.

I see at least three problems with **direct use** of the field of psychological theory and research as a knowledge base for CAI authors. One is the use of exotic jargon. Experimental psychology, the framework within which most of this work has been done is a mature science, having been engaged in intensive research activity throughout most of this century. As such, it has evolved a complex vocabulary and to benefit from reading the literature, the reader really needs the background knowledge that is gained in earning a terminal degree in psychology.

The literature of research on human learning that might be used by the CAI author is extensive. It contains many books, for example, the titles of which promise much that might be useful to the CAI author. However, the tables of contents give them away. They are almost invariably organized in terms of research categories that reflect both the specialized jargon of the field and the orientation of the research psychologist. Most of the literature was written to be read by other researchers and reflects that fact. Chapter headings such as "Paired-Associate Learning: Models and Processes", or "Serial Learning: The Functional Stimulus, Serial Position Effects, and Remote Associations", may have little meaning for the typical CAI author.

Lest we be accused of being too critical of researchers we should hasten to acknowledge that they are scientists, concerned with **how human beings learn** – not with how CAI courseware should best be written.

Even if the CAI author were thoroughly conversant with the literature of learning research they would still find it wanting, particularly in terms of completeness. There are simply a lot of things that are important to the CAI author missing from the literature.

Finally, even for those areas of CAI practice which are in the learning research literature, there is typically much required of the CAI author who would apply it to practice.

It should be expected that the practice of CAI courseware production will raise questions which are best answered by rigorous research techniques. The rapid increase in activity in CAI will certainly have a profound effect upon the activities of the research community, doing much to determine research priorities.

PREMISE III:

The structure of the CAI knowledge base is an important factor in determining its inclusiveness, usefulness and ease of refinement.

In order to be of maximum benefit, a CAI knowledge base must be acceptable to a wide variety of CAI professionals. Authors, programmers, trainers, researchers, professors, and others involved in CAI must be comfortable in using the knowledge base and participating in its maintenance. Therefore, the CAI knowledge base must be written in language that all will understand and its structure must be useful to all.

It may be useful to pattern the the CAI knowledge base after an "expert system" – a development of the artificial intelligence field. Indeed, it might eventually be possible to create an expert system in CAI courseware design. Some courseware generators seem to be moving in that direction. Following that line of thinking, a CAI knowledge base might have these components.

1. Rules of CAI practice
2. Rationales
3. Validation data
4. Normative data
5. Examples/nonexamples
6. Scientific research
7. Accumulated experience

To be maximally useful, a CAI knowledge base should be written in plain and simple everyday language – as free of jargon as possible.

The CAI knowledge base should be organized in terms of a **set of rules** of CAI practice. The rules should be expressed in simple declarative sentences whenever possible. If no experience or evidence exists to argue for or against a rule it should be thought of as a simple proposition, the efficacy of which remains to be proven – **but it should be written down.**

The rules of CAI authoring should be organized in a hierarchical manner so that the higher the rubric level, the more likely it is to form a complete set. Most likely the rules will never be considered complete. As practice changes, as new equipment capabilities are added, the accumulation of rules will surely change. However, the permanent state of incompleteness will not diminish the usefulness of the rules.

Examples of rules might include the following.

1. A blank to be filled in by students should occur near the end of a sentence.
2. The final frame of any lesson sequence should contain no prompting.

A number of elements should be associated with each rule. Perhaps the most important of these at this time is a rationale for practice. If there are competing practices they should all be included. This step is the best way to capture, in writing, the current state-of-the-art, as it is represented in practice.

The most meaningful information to support the rules comes from the practice of CAI validation. It matters little what the research base says or what anyone thinks, if validation data prove a rule incorrect.

Another element which would help establish the correctness of a rule would be the normative data gathered from CAI authors regarding their current beliefs and practices by using surveys and conducting interviews.

A given rule can be illustrated with examples and non-examples as a further indication of its meaning or application.

Scientific research should be designed and conducted which permits us to better understand and have confidence in the rules. Research should also provide insight into the relationships among rules and to their generalizability.

The collective experience of all those who use the CAI knowledge base can be fed back into the base and the base should thus increase in value and usefulness as CAI practice and CAI science mature.

Bibliography:

Burke, Robert L., CAI Sourcebook Prentice Hall, Englewood Cliffs, New Jersey, 1982.

A Computer Literacy Program for Elementary and Middle School Children and Teachers

Jack Halapin
Instructional Computer
Facilitator
Ridgefield Public Schools
Ridgefield, CT

56

Schools throughout the country are facing a brand new curriculum challenge, one for which they have neither experience or hard data upon which to base their decisions. At the heart of this challenge is a minute silicon integrated circuit surrounded by an array of electronic and electromechanical devices... a microcomputer.

Microcomputers, as we all know, are spreading like ground cover in elementary and secondary schools throughout the country. More computers have been installed in classrooms during the past 18 months than during the entire preceding decade. Last year alone, 70,000 units were sold to schools (*Business Week*, July, 1981). To be sure, we haven't reached the "computer in every classroom" prediction. However, that prediction may become a reality if schools can answer the questions: Why and what for?

The microcomputer is still a promise. Computer advocates, myself included, see an unlimited future for computers in schools. They can become the ultimate teaching aide, capable of individualizing and personalizing instruction. They can render management and accountability services for teachers inundated by mounds of paperwork. They can provide instructional simulations that heretofore were too costly or impossible to achieve, but most important they can become a conceptual tool for students who have been taught to program them to solve problems.

They can...but they won't unless the present generation of teachers and students become *computer literate* and thus be able to turn the promise and potential of the computer into reality. Then the questions of why and what for, will not be answered in terms of theoretical goals or potentials but will become obvious pragmatic realities, that give computers a place in schools because they help teachers and students to do the work of learning.

WHAT IS COMPUTER LITERACY?

In the broadest sense Computer Literacy is learning about and how to use computers. It comprises the knowledge and attitudes that will permit students to use computers in meaningful and productive ways throughout the course of their education and life. It includes teaching students how to operate and program computers and the practical use and application of computers as problem solving tools.

HOW CAN COMPUTER LITERACY BE ACCOMPLISHED?

The Computer Literacy scenario looks like this. A new curriculum is being written, not a K-12 curriculum but a T-12 curriculum. The "T" is the teacher who must be considered first. Many of us were around when the computer first came upon the education scene some 15 years ago. It never quite delivered all that it promised then. And the skepticism that "today's computer" will also fail to deliver is sincere and justifiable. However, today's school computer does not exist shrouded in mystery in some remote location. It does not cost the equivalent of ten teachers' salaries. The microcomputer, though a complete system containing all the components of its larger expensive ancestors, is an affordable desktop classroom tool which can be operated and controlled by teachers and students. Teachers then must be taught to use and control this medium.

The perceptions of the computer are wide and varied. Many teachers see the computer as non-humanistic, others view it as too difficult to use for the average person and some see it only as a toy for games. All these perceptions have some elements of truth to them but when the teacher learns control, then he or she will be able to recognize and shape the computer's positive aspects to meet his or her needs.

Along with the skepticism and perceptions also come the fears and these are very real. Tim Barclay of the Technical Educational Research Center in Boston outlines the fears in terms of questions teachers ask.

1) Will it (the computer) outsmart me?
2) Will the kids who use it outsmart me?
3) Will it replace thinking?
4) Will it replace me?

The answers to these questions are no on all counts, but the computer will make teachers change and change as we all know is difficult for any of us.

Thus this new curriculum will have to deal with the teacher first and recognize the positions from which teachers are coming.

How do you do it? How do you make a generation of teachers Computer Literate. You do it by Inservice and Support.

Inservice: We have a formal inservice course in Ridgefield (Introduction to microcomputers and BASIC Programming) designed to help teachers learn to use and control the computer. Hands-on experience is a must and classes are limited to no more than two teachers per computer.

My inservice program operates on five major principles. 1) Success from the first touch. I begin by having the teachers run a quality piece of software whose prompts and cues are well written and insure that they can proceed and answer questions without confusion. I do not use games but programs that either demonstrate the computer's capabilities or have curriculum content relevance to the group I'm instructing. 2) Proceed slowly (25 MPH). I want to have teachers understand and become successful users and am willing to take whatever time is necessary. 3) Get a little help from your friends. I usually have teachers work in diads, teams of two. Think back to when you were learning to drive. Remembering when to shift, depressing the clutch, watching the road, steering etc., all at the same time was quite a problem. So too it is helpful to have someone watch the screen and occasionnally remind you to press RETURN. 4) Mistakes can be corrected. I teach a language with a vocabulary and syntax that requires precision. Thus planning and attention to detail is stressed as a part of learning to program. But because we are using an interactive language and because we all make mistakes, the forgiving nature of the language is emphasized. That is, if you enter a command that is not executable, the computer will identify what it is it can't do and you simply correct your input and proceed. There are no records of your errors and no penalties. 5) Faith in the machine equals faith in yourself. When you reach that point where you understand that it is what you do that makes the computer do what it does, you have entered "electronic space" and will ultimately control it.

Support is also important. Someone must be available to solve problems, answer questions and fulfill teacher's requests. For example, at one school last year the teachers and principal expressed a desire to have some computer experiences for all their children, but many did not yet feel comfortable initiating a program. I set up a computer tour for the school and went into classrooms and provided lessons on computers for the children. In addition to limited history and computer concept lessons I taught access skills. Children were taught to load and run programs from tape or disk and the computer and software relevant to the grade was left in each room for a week with an option for a second week. Thus the instruction given directly to the children was also indirectly given to their teachers as well. By the end of the year all students and teachers had worked for at least two weeks with a computer.

We have now a formal inservice program for teachers as part of our total computer plan. It has three levels: awareness, orientation and programming. We do not teach programming with the objective to make teachers programmers capable of writing instructional courseware. This is unrealistic both in terms of the limited inservice time as well as the complexity of skills it takes to develop instructional programs. Our goals are simply understanding and control.

Once teachers understand the computer and begin to use it successfully with children they will be in the position to understand the capabilities and the limitations of this innovation and be more able to shape its future in the schools.

The computer was brought to my district to provide Computer Assisted Instruction (CAI) at the elementary school level. Most of our initial efforts and resources were and still are directed toward this goal. However, we soon recognized that there is an equally important goal which needed to be addressed, computer literacy for students.

Thus I began working with sixth grade students teaching them computer literacy which includes history, concepts, and programming skills. We have written a formal curriculum for grades 7-12 for Computer Literacy which integrates programming and problem solving skills into secondary mathematics classes. The experiments we are carrying out now are directed to introducing programming in lower grades. We have taught programming for the past two years as a part of the elementary gifted and talented program but now we are experimenting with programming skills in the regular classroom. We are running three similar pilot programs for all 6th graders at three elementary schools two of which are using BASIC language and the third PILOT language. The course covers approximately one full year and uses approximately 12% of the instructional time in the mathematics class. While there are some differences in the practice and application time, each class is instructed for approximately 45 minutes every two weeks by either myself or the classroom teacher on specific programming skills. The instruction is lecture and demonstration, followed by an assignment to use the particular programming skill in an application or practice problem. Students are then scheduled for 20 minutes of individual practice during the intervening time before the next lesson. Scheduling and monitoring practice time is handled differently in the three schools. In two schools a cadre of parent volunteers were trained to act as monitors and helpers and the students go to a "computer room" for practice. We are accomplishing this model, serving approximately 80 children at each school, with 2 Atari microcomputers in each computer room. In the third model we do not use parent volunteers and the students are scheduled to come to the teacher's classroom that houses the computer.

Evaluation is being undertaken primarily on a subjective level although we plan to employ a formal computer literacy assessment instrument at the end of this year. We are also looking at the tradeoff effect of the 12% of instructional classroom time and the 20 minutes practice time which now comes from individualized spelling and language arts student time. The programs work under our present curriculum structure although not without problems in schedule due to special assemblies, snow days, absence etc. We have been reasonably successful at insuring each child gets sufficient practice either on a scheduled or make-up basis.

Note that the computers are not dedicated to this program 100% of the time and they are available for use only in the afternoons since they are used for CAI in the mornings.

The Course Content is outlined below:

- The Computer & What I've Taught It To Do
- History
    - Fingers & toes to computers
- Concepts
    - Parts

Uses
Limitations
Access
Programming - BASIC
Commands
System - RUN, NEW, LIST
Program - PRINT, GOTO, LET,
INPUT, DIM,
IF....THEN,
FOR NEXT,
READ.... DATA,
GRAPHICS, PLOT,
DRAWTO

Just as with the teachers' inservice course similar principles also govern instruction for children. 1) Success from first touch. I want to insure that when the children work with the computer they achieve success. Here the parent volunteers have been very helpful. They serve not only to monitor the equipment but to make sure that the children don't get stuck or frustrated. They have been taught to help just that little bit to keep the student moving and to be there to answer questions. 2) Proceed slow (45 mph) - children seem faster and have less anxiety than adults. The real problem we've had has been the lack of keyboard skills which slows them down. You will have some students who will exceed the speed limit. My best advice is to get out of their way. If you can't help, find a high school student to come down and work with them. 3) Get help from your friends. In the beginning we have the students work in teams of two. Having someone with you to help watch the screen and confirm ideas in the beginning is helpful, but soon we want each child to have the individual experience of control. 4) Precision and planning are important. The computer is dumb, fast to be sure, but you have to teach it almost everything. We require that children come to the computer with a plan or a program on paper and when something they try doesn't work, get a hard copy so we can look at it later and figure out why. But we also point out that the program isn't *wrong*, it's just not *right yet*. 5) An instructional sequence is consistently followed for all lessons. (ONE COMMAND, A MODEL PROGRAM, AND A SUGGESTED EXPERIMENT OR MODIFICATION OF THAT COMMAND FOR PRACTICE.)

We believe that teaching children programming is teaching high level thinking. Teaching children to use a computer may be teaching them more about thinking than was ever possible before. Papert has pointed out in his book *Mindstorms* that in teaching the computer how to think children embark on an exploration about how they themselves think.

The results of teaching programming will first cause children to think about who controls the machine. They will make the abstract concrete by seeing their thoughts and ideas take shape on the screen. They will develop logical systematic thinking skills and will find out that, just as computers can't solve all problems, neither can linear thinking. They will learn that programs are never finished, but just as I am working on these ideas for an editorial deadline, programs have to be finished, perhaps to be reworked or enhanced later but never-the-less finished for now. They will also learn about errors, "bugs", and not think in terms of right or wrong but rather is it fixable. And finally, they will ultimately learn to use the computer as a tool to solve problems, gain information, conduct experiments and extend the reaches of their mind.

This is what Computer Literacy means, powerful ideas...wonderful potentials...and a challenge to schools but for only one generation. For if this generation of children can become computer literate, their children will come into a world in which the computer as a tool of learning will be commonplace.

Newtown, CT.
1982

USING THE MICROCOMPUTER TO APPLY DATA-BASED DECISION RULES FOR ASSISTING IN THE SELECTION OF INSTRUCTIONAL STRATEGIES FOR THE HANDICAPPED

Ted S. Hasselbring
Carol L. Hamlett

North Carolina State University

In recent years special education teachers and researchers have come to recognize the importance of collecting daily data on the performance of handicapped students in the teaching/learning process (Cooper, 1981; White and Haring, 1980). By collecting and graphing these data, teachers can determine when an instructional program is ineffective so that a change in the program can be made before it adversely affects the student. Unfortunately, the question of what type of change to make has been left largely unanswered.

In a study addressing this question, Haring, Liberty and White (1980a; 1980b) examined the effect of specific instructional program changes on student learning. From this data base, they developed a set of rules for assisting the classroom teacher in instructional decision-making. These data-based decision rules not only indicate when an instructional program should be changed, but also what kind of change should be made. In the third year of their investigation, Haring et al. found that teachers employing these rules made "successful" strategy changes (i.e., changes that produced an immediate increase in correct performance) 65% of the time. This compares favorably to "success rates" of 33% and 41% in the first two years of the study during which teachers did not use the decision rules (Haring, Liberty and White, 1980b).

While the use of these data-based decision rules has been shown to assist teachers in establishing appropriate instructional programs, the actual use of the rules in the classroom requires that the teacher carry out a number of specific steps. First, the teacher must develop a strategy for achieving each instructional objective, specifying the criterion level of performance and aim date for achieving that goal. Second, the teacher must collect three days of baseline assessment data and plot these data on semi-logarithmic graph paper. On the graph an "aimstar" is drawn at the intersection of the desired performance level and the aim data. A "minimum celeration line" is then drawn from the midpoint of the baseline data to the aimstar which indicates the minimum acceleration or deceleration in the student's performance necessary to achieve the criterion level of performance by the target date. Third, the teacher collects daily trial-by-trial data on each instructional program (e.g. the number of seconds the student holds head at midline in 10 trials) and transfers this information to the behavior chart on a daily basis. Finally, if the chart indicates that the student is progressing below the desired celeration rate, the graphed data are subjected to an additional decision rule flowchart which provides the teacher with suggested changes for the instructional program.

The tedious and time-consuming nature of this process has reduced the likelihood that this methodology will be used in classrooms. Few teachers have the time to employ the decision rule process on 8 to 10 instructional programs for one student, let alone an entire class. As a result, even though it is now possible to discriminate between successful and unsuccessful instructional strategies with some degree of certainty, the data-based decision rule process is seldom employed. One cost-effective solution to this problem is the exploitation of the microcomputer as a data collection, analysis, and feedback device in the decision rule process.

The authors have developed an integrated set of computer programs, entitled AIMSTAR, for use on a 48K Apple II Plus microcomputer equipped with two disk drives, printer, and monitor. AIMSTAR is designed to be used by persons with no computer programming experience or knowledge and allows the teacher or teacher aide to implement the decision rule process in the classroom easily.

The AIMSTAR Program

AIMSTAR is an interactive BASIC program that has been designed to store and retrieve classroom data as well as apply the data-based decision rule process as described by Haring, Liberty and White (1980a). Through AIMSTAR, the teacher is able to input and store daily behavioral data for student instructional programs. The computer then applies the data-based decision rules, reports instructional strategy suggestions, provides status reports on all programs on which a student is working, and optionally charts the data. The AIMSTAR program consists of four major components: (1) file definition, (2) data management, (3) charting, and (4) a report generator. A description of each of these components is provided below.

File Definition. The function of file definition is to enable the teacher to create and save on diskette descriptive information for each student's instructional programs. These files contain a description of the program objective, the desired level of performance and target date for achieving that performance, and the specific type of data to be collected (i.e., frequency or rate of correct/incorrect responses, duration of responding, or latency). These files may also include information on the instructional format such as antecedent conditions, definitions of correct and incorrect responses, and the contingent consequences for correct and incorrect responses. A master file for each student is automatically created which records all programs in a student's instructional plan, enabling the computer to process all programs in sequence, such as in report generation or data entry.

If and when it is necessary to make a change in an instructional strategy, the file definition program allows the teacher to edit only the portion of the instructional program that is being changed (e.g. contingent consequence) and if desired, calculates a new target date based on data previously collected.

Data Management. The data management component of AIMSTAR performs three primary functions--entry of data, editing of data, and application of the data-based decision rules. When entering data, the teacher can choose to enter data for a single program, or to have the computer prompt for entry of data for all programs automatically. After the teacher has verified that the data has been entered correctly, it is saved to disk. If six or more days of data have been collected, the computer retrieves from disk previously collected data as well as information on target date and performance level and applies the decision rules developed by Haring et al. Lastly, a printout is produced with student's name, date, name of instructional program and instructional strategy recommendations (e.g., whether to change a strategy and if so, what type of change to make). The data management program also allows the teacher to examine and, if necessary, edit data for any instructional program.

Charting. The charting program allows the teacher to display the instructional data graphically. The decision rules, as originally formulated by Haring et al., required daily plotting of data on six-cycle semilogarithmic graph paper before the decision rules could be applied. With AIMSTAR, the data entry program mathematically performs the necessary calculations for the decision rule process, thus charting is not necessary to apply the decision rules. Many special education teachers and administrators, however, have been trained to evaluate data by visual inspection, and the charting program allows them to view the data in a familiar form.

With the charting program, the teacher can choose either to present the data on the traditional six-cycle semi-log scale, or to utilize screen space more efficiently by having the computer adjust the axes. Likewise, s/he can choose to display the data for any desired time span, or accept program default values of the first and last days of data collected. In addition to this general plot of data, a "line of progress" showing current learning trend, can be displayed for the most recent six days of data. Any graph displayed on the screen can also be produced in a hardcopy format if a suitable graphics printer is available.

Report Generation. Student progress reports can be generated for any instructional program over any specified time period. Reports covering weekly, monthly, or yearly progress showing raw or summarized data can be displayed either on the video monitor or in hardcopy format. Once the report parameters have been specified, reports can be produced for all of a student's programs automatically by accessing the information in the master file on the student's data diskette.

Educational Use

The preliminary evaluation of AIMSTAR suggests that the educational implications of this program are significant. The data indicate that the AIMSTAR program has the potential of saving the classroom teacher significant amounts of time in record keeping over the course of a school year. This savings in administrative time should allow teachers to spend more time in direct instruction. For example, if five minutes of teacher time per pupil per day were saved using AIMSTAR, for a class of ten pupils, over four hours of teacher time could be saved weekly. In a classroom situation where teachers are expected to engage in instructional decision-making based on trial-by-trial data collection, AIMSTAR becomes a highly economical system.

The preliminary data also suggest that teachers are more likely to incorporate decision rules in their teaching using AIMSTAR. If this is true, then the AIMSTAR program could have a positive effect on the education of large numbers of handicapped students. One would posit that as a result of AIMSTAR, a greater number of appropriate instructional strategies would be employed by classroom teachers resulting in a concomitant increase in the development of their students.

In summary, while an extensive evaluation of the AIMSTAR program is certainly necessary, all early indications are that this program will have a positive impact on the field of special education. Through the use of AIMSTAR, the need for teaching intuition and judgement will not be replaced; instead they will be supplemented with empirical data and analytic procedures. It is hoped that through the use of this technology, teachers will be able to determine with much more success those instructional strategies which result in positive student gains.

References

Cooper, J.O. 1981. Measuring Behavior. Columbus, Ohio: Charles E. Merrill Publishing Co.

Haring, N.G., K.A. Liberty, and O.R. White. 1980a. "Data-Based Decision Rules for Instructional Programs." Unpublished monograph. Seattle, Wash.: Child Development and Mental Retardation Center, University of Washington.

Haring, N.G., K.A. Liberty, and O.R. White. 1980b. "Rules for Data-Based Strategy Decisions in Instructional Programs: Current Research and Instructional Implications." In Sailor, W., B. Wilcox and L. Brown, eds., Methods for Instruction for Severely Handicapped Students. Baltimore, Md.: Paul H. Brookes.

White, O.R. and N.G. Haring. 1980. Exceptional Teaching (2nd ed). Columbus, Ohio: Charles E. Merrill Publishing Co.

A PROBLEM SOLVING AUTHORING SYSTEM

by Ted Perry

Developing Computer Assisted Instruction lessons seems like an excellent idea until one understands the complexity of the task. Granted there are educators with excellent programming skills. There are also educators with the time and energy to learn to become accomplished programmers. There are even some educators with the time to learn programming and the time to devote to designing excellent teaching programs. However, most educators are neither professional programmers nor overwhelmed with free time.

The current crop of commmercially available Computer Assisted Instruction programs is not adequate. While there are good educational programs in many areas and excellent educational programs in a few areas - notably elementary level math computation - there are few broadly based curricula.

To help fill the gap between the programs that are available and the programs that are desired, BLOCKS II * author language was developed. BLOCKS II author necessitates no programming expertise on the part of the teacher. BLOCKS II literally walks the teacher a step at a time through the process of building a student lesson. The teacher is able to type the text of a lesson noting correct answers, expected incorrect answers, and appropriate feedback. BLOCKS II also makes full use of the graphic capabilities of the Apple II computer by giving the teacher a library of graphics and the vehicle to place those graphics where desired. Lesson planning and automatic data collection are provided.

BLOCKS II author provides a fertile field for the development of tutorials using multiple choice and fill-in questions; BLOCKS II is currently being used by over 1100 school districts throughout the U.S. and Canada. But while BLOCKS II has formidable capabilities, it also has limitations. BLOCKS II curricula is presented in either a random or sequencial order, but each lesson is presented exactly as written by the teacher. In other words, the questions in BLOCKS II cannot themselves contain variables. A variable-based authoring system is needed to allow the teacher to more easily construct an assortment of lessons to teach a single concept.

MATHWORD, the beginning of a new generation of author languages, was developed to fill this need in the areas of problem solving and mathematics. MATHWORD provides a step in the direction of a "general problem solving method." The students have the opportunity to become more aware of the similarity among approaches to problem solving and the steps which must be followed to solution.

MATHWORD author allows the teacher to develop generic problems and lessons, provide a range of variables, and present lessons to the student. To understand this authoring system, the structure which is needed for a problem solving lesson must be explored.

A word problem can be seen as a format which contains some constants in the form of words, numbers, and formulae. Students need to be presented with each type of word problem several times; each presentation should use different words and different numbers so that students learn to generalize beyond the immediate limited context. A generic word problem contains a set of words, numbers, and formulae needed to develop this series of lessons. A generic word problem authoring system - MATHWORD - allows the teacher to define the ranges within which the system creates a wide variety of lessons. For example:

John has 3 white eggs, Tom has 4 brown eggs, and Suzie had 7 white eggs, but the cat ate 1.

or

Given a gaseous mixture of 4 moles of oxygen and 9 moles of hydrogen under standard temperature and pressure, and given the introduction of a flame:

SOME POSSIBLE QUESTIONS:

How many eggs do John and Suzie have together?

How many more eggs does Tom have than John?

Which color of eggs is predominant?

or

What are the products by quantity and quality?

How much more oxygen is needed to combine with all the hydrogen?

How much heat is given off in the reaction?

The strength of MATHWORD is in the power, flexibility, and generativity it gives to the teacher. MATHWORD makes it possible to quickly and easily create a series of lessons which teaches to a specific objective. MATHWORD provides branching to teacher- specified "help" sequences which are determined by the type of error the student makes. MATHWORD even provides an on board calculator, if the teacher so desires. MATHWORD is currently in a field testing phase. Persons interested in participating in the field testing are encouraged to contact the author.

* BLOCKS II
Ted Perry
San Juan School District
6141 Sutter Street
Carmichael, Ca. 95608

** MATHWORD
APPLE FOR THE TEACHER,
2331 Rainbow Ave,
Sacramento, Ca. 95825
Atten: Ted Perry

68

COURSEWARE CRITERIA FROM AN INSTRUCTIONAL DESIGN PERSPECTIVE

M. D. Roblyer
Director, ICON Enterprises

Introduction

Perhaps one of the most encouraging trends in the instructional computing field today can be seen in the emphasis which many organizations are placing on evaluating courseware materials before making any major purchases. It also seems significant that these evaluation activities have shifted the early emphasis on addressing technical capabilities of programs to that of certifying instructional quality of courseware. These kinds of evaluation activities may be viewed as an indication that educators are rapidly developing an informed perspective both on the appropriate role of courseware and of their reponsibilities as consumers of instructional computing products.

Although the ideal situation would be a field trial of courseware for the purchase of gathering evidence of instructional effectiveness, this is rarely feasible. Evaluation efforts are usually carried out by selection committees which review courseware according to certain criteria identified beforehand. There is still considerable speculation, however, as to which criteria are most useful in predicting courseware quality and effectiveness with students.

One way to consider the characteristics most desirable in courseware is to view them from an instructional design perspective. The term "instructional design" as used here refers to both the systematic methods used to plan and develop the materials and to the plan and product resulting from such methods. An instructional design process is employed to assure that instruction is as effective and efficient as possible in bringing about learning in a specific group of learners. By examining the elements of a sound instructional design process and product, it is possible to identify some fairly specific criteria which describe good instructional design for a courseware product. This paper will list and describe groups of criteria derived through such an analysis, as well as some problems and concerns in attempting to use these criteria to guide an evaluation effort.

A Summary of Instructional Design Criteria

Two aspects of design concerns are considered integral to the success of an instructional package. The first pertains to the selection and structure of activities used to cause learning, and the second has to do with characteristics provided to help manage the instruction, that is, to make clear to both the teacher and to the student how to use the materials. Both of these kinds of concerns bear on the instructional effectiveness of the package, and thus, both are reflected in the criteria described here. For the purposes of this brief paper, the criteria will merely be listed. They will be described more fully in the presentation.

Area I: Specification of Skills and Learners. The first step in an instructional model is usually identifying the target population and the behaviors which they are to exhibit as a result of the instruction. This should also be the first consideration in a courseware evaluation. Locating statements of objectives and target students helps not only in evaluating the rest of the package, but also makes it easier to see how the courseware fits in with other skills being addressed in the classroom. Some questions to ask include:

1. Are objectives given (either in the courseware itself or in a Teacher Manual) in the form of statements or examples? Are they in terms of what the student is to do, and not in terms of what the teacher or courseware is to do?

2. Do skills and/or information being taught represent educationally significant items? (Items which are currently taught or should be taught in school curricula?)

3. Do objectives reflect behaviors which are as close as possible to those required in real life activities? (Not artificial activities designed so that the computer can accomplish them?)

4. Is a target population specified as to level, characteristics and assumed prerequisite skills?

5. Are the skills appropriate for the target population, according to what is known and widely practiced as to the difficulty of the learning tasks and students' readiness to learn them?

Area II: Sequencing and Grouping of Skills and Activities. Next, an instructional design model usually calls for an analysis of the desired behaviors and of the target students to discover an optimal order for presenting the lessons or lesson parts. This aspect pertains more to a courseware series and to packages with more than one skill than it does to a single-skill game or drill. Concerns in this area include:

1. Are the lessons/skills organized in a way appropriate to the type of content and most usable by the students and teachers, i.e.:

a) In a hierarchical, easy-to-difficult order (for many math and other skill-related areas), or

b) Into appropriate content topics or groups (for many information-type topics)?

2. Is it easy to see where in the continuum to place a student who is at a given skill level?

3. Is a scope and sequence provided in the documentation to aid in placement and evaluating lesson content?

Area III: Assessing Performance. Although courseware does not have to have mastery-type tests, it is necessary that there is some way for both the student and teacher to tell if satisfactory progress is being made.

For mastery pretests and posttests, one should be able to answer the following:

1. Are pretests included for a specific purpose, either to diagnose need areas or to allow "testing out" of the materials?

2. If diagnostic, are test items or parts matched to specific parts of the courseware, so that students may go directly to instruction required?

3. Are mastery levels set at appropriate criteria?

For both mastery tests and for practice exercises:

4. Is there a sufficient number of items to assess how much the student has learned?

5. Are the items closely matched to both objectives and instruction?

6. Do tests have:

a) Adequate directions?

b) Summary of performance at end?

7. Are item formats appropriate to assess the behaviors? (i.e. multiple-choice vs. fill-in vs. matching, etc., as needed?)

8. Are response modes appropriate for the age and ability levels of the students?

Area IV: Instructional Activities. These are the most important items to consider in any evaluation, since they deal with the "meat" or substance of the courseware. These are probably also the most difficult aspects to describe in specific terms. Fortunately, it is often easier to spot an ineffective design in a given instance than it is to describe an effective one in general terms. The following are some aspects which can help identify any problems:

1. Is the difficulty level of the instructional activities appropriate for the target students in:

a) Kinds and levels of examples and explanation?

b) Readability of text?

2. Is the instructional strategy suitable to teach the content/skills to the target students? Consider these components:

a) Adequate directions?
b) Explanation clear and concise?
c) Adequate number of examples?
d) Provision for gaining attention, recalling prerequisites, and other instructional events, as necessary?
e) Provision for reviews of content, as necessary?
f) Adequate number of embedded practice items?
g) Adequate and appropriate feedback given on incorrect answers to practice items?
h) Adequate and appropriate reinforcement for correct answers?

3. Is the length of each lesson an appropriate sized "chunk" for the target students to assimilate at one time?

4. Is the lesson under the appropriate degree of user control (i.e. times only as necessary, and then with enough time for read frames and to answer and study frames, as needed)?

5. Are there off-line support materials provided as needed, and do they meet the criteria described above for on-screen materials?

Area V: Teacher Management. For courseware to be optimally effective with students, it should make provisions for easy implementation into the classroom. Teachers should readily be able to see where the materials fit in meeting the needs of students and how to obtain and interpret the results of computer-based instruction. Evaluators should address these questions:

1. Is there a Teacher's Or User's Manual and does it provide adequate explanation of how to use the package, both technically and educationally?

2. Is there an on-screen or paper management system provided to help keep track of student progress?

3. If the management system is on-screen, does it have options to:
a) Add/delete/list students
b) See reports of student progress
c) Get information on specific areas of problems, both for an individual student and for the whole class?

4. Does the Teacher's Manual tell:
a) What is required to teach students how to use the courseware?
b) How to use the management system?
c) Simple trouble-shooting procedures to handle courseware idiosyncracies?

Using Instructional Design Criteria

Several concerns are immediately apparent in attempts to evaluate courseware according to criteria derived from instructional design variables. First, and probably most important, the criteria are only useful to those who can identify good and bad examples of each of the items. This means that the courseware evaluation team must be well-prepared for their task by seeing such examples and receiving training in the qualities of good and bad designs. There should also be a content expert on the team for each of the areas of materials reviewed.

Other problems which can hamper evaluation efforts aimed at instructional design concerns include the following. Most courseware:

- Has not been systematically designed. Since this kind of development requires special expertise and a considerable time investment, there have been few development projects to date which have employed such techniques.

- Is not a complete instructional system. An instructional design process is usually done to define and develop the needs of a totally self-contained instructional package. Much of the courseware done to this time is drill and practice and includes only some of the instructional events and components of such a package. It is usually assumed that many of the events will take place outside the package in other materials and by the teacher.

As is usual in attempts to make courseware evaluation and selection a more pragmatic process, the criteria described here tend to raise more questions than they answer. However, even though it may be difficult to apply such criteria in a rigorous way, it still seems essential to consider these aspects in any review of courseware. Focussing on these aspects, rather than on appearance-related elements, will help assure that the courseware selected becomes an integral part of classroom instruction, rather than just an interesting diversion.

AND THE TWAIN SHALL MEET: COMPUTERS AND THE TEACHING OF ENGLISH

Robert Shostak

Director, Institute for Educational Technology

Many English teachers see the computer only as a useful tool for the mathematician, scientist, or the administrator. Consequently, they dismiss altogether the idea of using computers in their own work. Others see computers as a kind of robot-like creature, a view perpetuated by many Hollywood science-fiction films. These teachers simply do not take computers seriously at all. Still another group of English teachers - the tough old birds who have seen it all - nod their heads knowingly, while muttering (just loud enough for you to hear) something to the effect that it will not be too long before computers will be gathering dust in the closet alongside all the opaque projectors, television sets, phonograph players and other assorted once-heralded pieces of "modern" technology.

At one time or another I probably held some of these same views regarding the use of computers and the teaching of English. Now, however, I do not think any English teacher can afford to subscribe to this kind of thinking. Because of the pervasiveness of the computer in every aspect of our public and private lives, we can no longer afford to ignore its impact. More importantly, we can not afford to ignore the potential it holds for education.

The use of technology and the need to understand its effects on society has not gone unnoticed in higher education either. Colleges and universities are looking for effective ways to integrate knowledge of technology with humanistic, ethical and even theological considerations found in college curriculums. A unique program at Saint Louis University called Man, Technology and Society offers an English course called Madness in 20th Century Literature. As you might expect, the goal of this program and other similar programs is to demonstrate the importance of understanding the critical interplay of technology and the humanities in our rapidly changing society.

However the decision to use computers in classrooms is not going to be made solely by English teachers. The integration of any new technology into existing systems is a difficult process even under the best conditions. It is no secret that public schools are beset by more problems than they deserve.

Two of the problem areas which professional educators are going to need to rethink very carefully if computers are going to be used effectively in our schools, are those dealing with how children learn and the curriculum.

Seymour Papert, in his book, Mindstorms: Children, Computers, and Powerful Ideas, says that the teaching profession believes the way to increase learning is to improve teaching. This view, he believes, simply creates "...an artificial and inefficient learning environment that society has been forced to invent because its informal environments fail in certain essential learning domains, such as writing or grammar or school math." Papert suggests educating young people to remove the artificiality from the learning environment and increase efficiency of learning based on the way a child learns to talk. He indicates that this complex learning task takes place in a natural environment without the benefit of "deliberate" or "organized teaching". Papert believes the computer can help create a more natural environment in which children will learn school knowledge as successfully and painlessly as they learned to talk.

If we adhere to the systematic

Florida Department of Education

"artificial", teacher-centered approach we rely on so heavily today, then the computer will become simply another instructional aide not much different from a tape recorder or overhead projector. However, if we begin to recast our thinking about the classroom environment and emphasize learning rather than teaching, then the computer will become the medium through which students develop intellectual prowess and create knowledge for themselves.

Although most recent publicity has focused on the schools' inability to teach basic skills, teachers know that the real weakness in the curriculum is in the area of problem solving. David Moursund, in his pamphlet School Administrator's Introduction to Instructional Use of Computers, presents an incisive analysis of the problem.

He accurately points out that most curriculum devoted to problem-solving focuses on carrying out a plan for the solution to a problem based on a formula or other recipe-like approach. Little time is spent on trying to understand the problem, formulate a solution, or interpret results. Moursund believes that computers can handle the formula solution portion of the problem-solving process better than people can. The computer can process information much faster and certainly with a higher degree of accuracy than humans. He suggests that our "overall curriculum needs to change, to reflect the role that calculators and computers best play in problem solving." Taking this direction will "decrease emphasis upon the routine and rote skills of carrying out a plan to solve a problem" and increase emphasis on the higher level skills of problem-solving.

More specifically, if English teachers are not going to be able to use the computer to help students develop higher order thinking skills, I believe they are going to reject its use in the classroom. English teachers want students to develop their skills in analysis, synthesis, and evaluation. These are the critical skills required for competency in the areas of literature, language, and communications.

In addition to the need to recast their thinking about the nature of curriculum and the way students learn, English teachers must consider their own professional development. Two pieces of reality that English teachers face at this time are lack of knowledge about computers and the rapid rate at which the technology is advancing. The solution to the first problem is education. As quickly as possible, English teachers should and must demand effective computer literacy training. With a sound understanding of how the computer functions and what it is capable of doing as a learning tool, teachers can then begin to conceive better the possibilities inherent in using computers in the English classroom.

The solution to the second problem is not as simple. Computer technology is advancing at such a rapid rate that I sometimes get the feeling that if I don't step out of the way it will run right over me. What could happen is that as teachers we might find ourselves at the mercy of those who control the technology but do not necessarily understand what is really important in the teaching of English.

As professionals, then, English teachers will have to play a major role in establishing guidelines for development of computer courseware for the language arts. More specifically, English teachers will have to help define the curriculum content and objectives which are taught best by computers. They will have to suggest innovative uses of this new technology and identify appropriate areas of research. Finally, English teachers will have to assume a major part of the responsibility for making other English teachers aware of how computers can be used effectively in the classroom.

The use of computers in the classroom by English teachers is presently very limited. It is limited both by the lack of equipment and availability of good computer courseware. Teachers fortunate enough to have computers are using them for the most part as teacher aides. For example, packaged programs produced by well-known publishers of educational materials and some recent new-comers are being used to teach spelling, punctuation, parts of speech, and vocabulary.

Other computer programs provide drill and practice exercises in these curriculum areas so that students may develop and extend their skill. A distinct advantage of using the computer in this way is that it provides some students with truly individualized instruction and an accurate up-to-the-minute record of their progress while freeing the teacher to provide other students with learning opportunities which require close teacher/student interaction.

A number of imaginative ways have been found to use the computer to help students better understand and appreciate poetry. Using simple programs to generate lines of poetry in specifically structured patterns, such as a haiku, some English teachers have had great success in teaching about the creative process. Students who understand how the computer generates poems are quickly inspired to create their own poetry using the computer program as a guide. The heuristic nature of this activity brings students face to face with problems all authors experience in producing a piece of creative writing. They must pay careful attention to planning ahead, diction, spelling, punctuation, and grammar. More importantly, they are quickly and easily able to share each other's work to engage in critical exchanges which lead to a greater understanding of and deeper appreciation for poetry as well as the creative process.

Some teachers have found that the computer's ability to function as a word processor is extremely helpful in teaching composition. This capability allows students to compose on the computer's keyboard; save their work so that it may be recalled at a future time; edit (at the touch of a key) every aspect of what has been written, including the shifting of sentences and paragraphs from one location to another; and print out a finished product without having to be concerned about spacing, pagination, headings or margins. In other words, revision, so necessary in the process of learning how to write well, yet so distasteful to students, can be so simplified by the word processing power of the computer that students who have tried it actually enjoy it.

Another innovative use I have discovered utilizes computer adventure games similar to the familiar Dungeons and Dragons, to teach certain aspects of literature. For example, students find it very easy to understand the literary elements pf plot development and characterization once they understand how an adventure game employs similar techniques in its format. The adventure game is also a great motivational tool for developing students' creative thinking powers. It makes them aware of their creative ability --a rich but untapped resource in many students--and more importantly, provides an area for students to produce new ideas and new creations of their own.

Lastly there is the powerful management capability of the computer which English teachers are finding especially helpful in their classroom. Many teachers no longer need to be concerned with the clerical function associated with administering, scoring, and recording test results. The computer as a grade book is accurate, efficient, and remarkably swift in calculating final grades and providing the teacher with complete printouts of class assignments, student rankings, and a variety of statistical information valuable for instructional decision-making.

Although many educators are not yet aware, we are, without a doubt, in the midst of a technological revolution which is already having its effect on a number of English classrooms around the country. The National Council of Teachers of English has anticipated the rapid changes which are going to take place by appointing a standing committee on Instructional Technology to begin to develop a specific plan to assist English teachers in the task of integrating technology into existing English programs.

If educators fail to meet the needs of society as it moves inexorably into the age of advanced technology, they may find themselves in the same shaky position the American car industry finds itself today. The failure of Detroit to make the successful transition from the technology of electron/mechanical production to the technology of the computer has cost them the number one position in auto manufacturing throughout the world and a corresponding loss of billions of dollars. American car manufacturers may recover from this setback. The education establishment may not get a second chance.

Language arts teachers in the elementary schools and English teachers in secondary schools have successfully met the educational challenges of past decades. Their dedication, curiosity, and strong sense of responsibility will once again serve them well as they strive to seek ways of transforming the English curriculum to reflect the changes certain to be brought about by the technological revolution of the '80's. The old antipathies separating technology and the humanities are rapidly crumbling. I am hopeful that English teachers, in particular, will assume a leadership role in what I see as a new rapprochement between these powerful forces in our society. And (as a result) the twain shall meet: computers and the teaching of English.

References

Toffler, Alvin. The Third Wave. New York: William Marrow and Co., 1980.

Papert, Seymour. Mindstorms: Children, Computers, and Powerful Ideas. New York: Basic Books, Inc., Publishers, 1980.

Moursund, David. School Administrator's Introduction to Instructional Use of Computers. ICCE, Eastern Oregon State College, 1980.

DISTRIBUTED DATA PROCESSING: GENERAL CONSIDERATIONS AND A USER EXPERIENCE

Amjad Umar
Manager of Systems,
Wayne County Intermediate Schools District,
33500 Van Born Road,
Wayne, Mich, 48184

ABSTRACT

In recent years, a large number of educational as well as non-educational organizations have either considered or are in the process of considering Distributed Data Processing (DDP) services. This paper addresses the problems of choosing, installing, operating and managing the DDP systems. The first part of the paper forms the basis for analysis and evaluation of various options. In the second part, the currently available research as well as commercial approaches are reviewed and categorized. The third part briefly describes the DDP system at Wayne County Intermediate School District, recounts our experience and uses this implementation to illustrate various choices and trade-offs.

1 OVERVIEW OF DISTRIBUTED SYSTEMS

1.1 INTRODUCTION

In the past decade, the computing and communication technologies have advanced dramatically. The concept of Distributed Data Processing (DDP) has emerged due to a marriage between these technologies. (Some authors, for example Kleinrock <KLEI76a>, refer to this as a "shot-gun" marriage.) While the definitions of DDP vary widely (see for example the fifty definitions collected by Bagley <BAGL79a>), the concept basically allows a collection of files and programs to be accessed and managed on a network of geographically distributed computers. This concept appeals to many organizations mainly because it allows a user to perform some functions locally (perhaps on an inexpensive processor) and use the facilities of remotely located processors for other functions. Consequently, a large number of academic, financial and government enterprises are migrating towards DDP. However, once the decision to adopt DDP has been made, the organizations face serious challenges of choosing, installing, operating and managing the DDP systems - the focal point of this paper.

A plethora of articles, papers and books have been written, and are being written, on DDP. It is extremely difficult for a prospective user to read, understand and utilize this information effectively due to conflicting terminology, esoteric research details and lack of practical insight. The primary objective of this paper is to present a synthesis of the applicable literature and illustrate the relationship between a practical implementation and published material. To reduce the size of this paper, we refer heavily to the literature listed in the bibliography for details. The bibliography lists major references and is organized in four convenient sections. The reference BAGL79a, for example, indicates that the paper is listed in section a of the bibliography.

We start out by presenting a framework for analyzing and comparing DDP systems. This framework subdivides the data processing facilities into application systems, support systems, and organizational systems. In section 2, we use the framework to present the various challenges and approaches in terms of the application systems, the support systems, and the organizational systems. Section 3 describes the DDP implementation at Wayne County Intermediate Schools and shows how this implementation can be analyzed in terms of the proposed framework. Section 4 concludes this paper by summarizing the main points.

1.2 A FRAMEWORK FOR ANALYSIS

A data processing facility is a collection of the following three types of systems:

- An **application system** which constitutes the user defined data (the database) and the software instructions (the programs) to manipulate the database. Examples of application systems are payroll systems, air line reservation systems, student registration systems and the like. As can be seen, an application system may be a simple report program or a large and complex command control system consisting of several hundred programs.

- A **support system** which provides the hardware as well as the software resources needed by the application systems. The hardware resources consist of the computing and I/O processors, storage devices, communication devices and the like. The software resources consist of the operating systems to manage the hardware resources, the database managers to manage access of data by programs, the communication managers to coordinate the remote access of information and several other software "packages". As can be expected, for a given application system a large number of support system configurations can be proposed. Keeping this in mind, it is difficult, if not impossible, to design an "optimum" support system configuration for a large number of concurrent applications. Due to this reason, many organizations choose the support system configuration to best suit the critical applications. Once this is done, the rest of the applications are designed within the constraints imposed by the existing support system.

- An **organizational system** which consists of the personnel (operators, programmers, analysts, managers) and the procedures (standards, access codes, operating instructions) needed to design, install, maintain and operate the application and support systems.

It should be emphasized that the core of the data processing facility is the collection of application systems - the support and organizational systems basically accommodate the applications. This suggests that the data processing facilities can be categorized in terms of the location of application system components. Thus a Distributed Data Processing (DDP) facility can be viewed as an environment in which components of an application system (data, programs) can be distributed to and accessed from different geographical locations (nodes). Similarly, a Centralized Data Processing (CDP) facility would restrict the location of application systems to one location.

It should be noted that the above definitions automatically enforce appropriate support and organizational system structures. For example, to facilitate DDP, support system and organizational system components (hardware, software, personnel) must exist at several nodes. However, we are not restricted to specific support system and organizational system structures. For example, in a CDP environment, the organizational system components can still exist at several locations (software can be developed at one location and used at several locations or vice versa). A management model proposed by Rockart <ROCK77c> deals with the interrelationships between the various systems in more detail.

2. GENERAL DDP CONSIDERATIONS: CHOICES & APPROACHES

Design, development and management of DDP systems involves several challenges. As can be expected, a wide variety of approaches exist to meet these challenges. The framework introduced above allows us to discuss the challenges and approaches in terms of the application, support, and organizational systems.

2.1 DISTRIBUTED APPLICATION SYSTEM CONSIDERATIONS

The distributed application system choices can be discussed in terms of distributed databases and distributed programs.

The design of **distributed databases** involves the major decision of allocating data to nodes. The database can be allocated to nodes uniquely or redundantly. Theoretically a given data file should be replicated to all nodes so that a user at any node can access it. However, this presents the problems of increased storage costs, maintenance of duplicate data and possible security violations. Allocation of data also depends on "data specificity" which is defined as the accuracy with which the origin of data access can be predicted. For example, data is 100% specific if one can always predict the nodes from which the data accesses originate. On the other hand, data is 0% specific if the access for data cannot be predicted. Obviously, the data can be allocated to requesting nodes for 100% specific and kept at a central site for 0% specific situations. However, a wide variety of options exist in between. Thus the data allocation decision presents several alternatives. The classic database allocation problem is: given data retrieval and update frequencies from various nodes, determine the allocation of data to nodes which minimizes the communication and storage costs. This problem was first addressed mathematically by Chu <CHU69c> in 1969. Since then over twenty papers on data allocation have been published. The interested reader is referred to section C of the bibliography for applicable examples of mathematical solutions.

Design of **distributed programs** is influenced by the software replication problems, i.e. the same program may need to be installed and operated on several, and in many cases heterogeneous, systems. This complicates the well known problems of software cost estimation <ARON69d, BOEH74d, PUTN78d, PUTN76d, WALS77d>, software development <KEND77d, TEICH77d, BOEH73d, PARNA72d, BELA77d> and software management <PRES75d, GANS76d, KEND77d, THAY75d, DALY77d>. Special programming languages are needed to address these problems (see for example Feldman <FELD79b>). An impressive survey of programming languages for DDP has been given by Silberchatz <SILB80b>.

2.2 SUPPORT SYSTEM CONSIDERATIONS

The major problems in distributed support systems are concerned with the design, evaluation and testing of software components. We describe these problems in terms of the distributed database, distributed communication, and network resource management.

A **distributed database manager** is responsible for the access and maintenance of distributed data. The major challenge in developing a distributed database manager is to provide update synchronization and data model support. The problem of update synchronization arises due to the existence of duplicate data. The main problem is: how can the duplicate data be synchronized to provide a consistent view of data and also allow maximum concurrency? Rothnie & Goodman <ROTH77a> give a simple yet elegant treatment of this problem. Several aspects of these problems are still unsolved, however a few experimental distributed database managers have been developed by various R & D establishments. Examples are, SDD1 by the Computer Corporation of America <ROTH77b>, INGRES by the University of California at Berkley <STON75b> and 1DER at the Bell Research Laboratories <LIEN78b>. Several researchers <GARC79b2, BERN78b, ELLI77b, THOM78b> have worked on this problem and research in this area is very active.

The situation of data models for distributed systems is also not pleasing. The main data modelling problem is to provide views of data to programs for data manipulation. Data models fall into three general categories: a) one level models which pass the actual physical organization of data to programs b) two level models which separate the physical view of data from program's view of data and c) three level models which introduce an intermediate conceptual view of data between the physical and program's view of data. Basically, the conventional flat files, say ISAM, support one level models, systems like IMS support two level models and future relational systems support three level models. Details of data models are given by Senko <SENK77d>, Tschritzis <TSCH78d>, Chen <CHEN77d, CHEN77d2>, Teorey <TEOR80d> and Fry <FRY78d>. Description, storage and access of data models in distributed systems involves very detailed technical problems due to the differing data models at several nodes. The interested reader is referred to Adiba <ADIB78a>, Mariani <MARI79a> and Rothnie <ROTH77a>. In short, appropriate data models for distributed systems are not are not available.

The status of **distributed communication management** is quite encouraging. The major problem in distributed communications management is to facilitate addressing (identification of each node), routing (carrying of messages between nodes), session control (signon/signoff) and network topologies (interconnections between components). Unlike the distributed database management problem, several successful mechanisms have been established and are available commercially. For example, available distributed communication managers at present are IBM's SNA <IBM75b, CYPS78b>, Digital's DNA <WECK76b> and Burroughs BNA. Thurber <THUR80b> gives an excellent comparative overview of various approaches. The books by Doll <DOLL78a> and Davis <DAVI79a> also treat this subject at length. Due to the differences between the various implementations, a unification effort has been launched by the ISO/ANSI committee. The seven layer reference model proposed by the committee has been described by several authors <BACH78b, DESJ80b>.

The **distributed operating systems**, commonly termed as Network Operating Systems (NOS), attempt to provide remote processor sharing, failure handling and recovery. The objective of such systems is to detect overload or failure of remote processors and route work automatically to other processors in the network. Example of a NOS is the UNIX system <HOLG78b>. Considerable work is being done in this area (see Forsdick <FORS78b> for details>.

2.3 DISTRIBUTED ORGANIZATIONAL SYSTEMS

Besides the technical problems presented in the above sections, several organizational and management problems need to be addressed in DDP. For example, what organizational structure is best suited for DDP, how the people can be trained, how the human interfaces can be defined, how can adequate pricing policies be developed, what type of standards and policies should be enforced etc. Since support of DDP systems is multidisciplinary, matrix organizational structures, which allow teams across departments, are perhaps the best. Unfortunately, very little experience/knowledge in this crucial area can be found in open literature. The papers by Smith <SMIT80b> and Lorin <LORI80b> provide some insight.

3. AN EXAMPLE OF DDP IMPLEMENTATION

So far, we have mainly described the generalized DDP considerations based on available literature. We will now describe our own experience with DDP and attempt to show the relationship between generalized systems and the specific implementation of DDP at the Wayne County Intermediate School District. The Wayne County Intermediate Schools District operates a data processing facility to serve the information processing needs of about twenty-eight school districts in the Detroit area. The facility was initially conceived in 1973 and was implemented as a centralized data processing environment with the following major application, support and organizational systems:

1. The centralized application systems consisted of student services, finance, payroll and miscellaneous testing and statistical systems. All these systems were strictly batch oriented.

2. The centralized support system consisted of an IBM-370-158 computing system (host) operating under the OS/MVT operating system. The centralized computing system was accessed by remotely located users through remote job entry workstations (IBM 3780 terminals) connected to the host in a star type network topology.

3. The organizational system consisted of six staff members (operators, systems programmers and application consultants), located at the central facility. The application systems were, however, developed at a remote site, the RAMS Cooperative Services. Thus, the organizational system was somewhat distributed.

Between 1973 and 1977, the centralized data processing facilities saw the following changes:

1. The student services application system was rewritten to provide IMS database support.

2. The OS operating system was replaced with an MVS operating system and the IMS database management system was installed.

3. The size of the organizational system grew to about 25. The staff was organized in three major groups: operations, systems and applications support.

In 1977, the decision to migrate from CDP to DDP was made to: i) offload the host processor, ii) provide local access/maintenance of some data and iii) position for future applications like word processing. The main thrust of the DDP migration is to replace the remote job entry workstations with minicomputers which can run some applications locally and also interface with other applications at the host. An initial analysis indicated that the following considerations should be taken into account:

1. Application systems: The allocation problem is simple because the data is 100% specific and because each node only accesses its own files. Thus the data can be allocated to the requesting node. However, a master copy of files is needed at the central site for auditing

and control purposes. This introduces data duplication and creates update synchronization problems. Since all nodes run essentially the same applications, the distributed programs need to be replicated at each node.

2. Support system: Since no distributed database manager is available, the update synchronization problem must be handled by special programs and only flat files (one level data models) can be supported. Due to the existing star type network topology, the communication manager must be centrally controlled. At present there is no need for network operating systems, however it is important that the distributed processors have suitable hardware/software interfaces with the existing support system.

3. Organizational systems: Due to the impact on end-users and complexity of the migration task, proper project planning and control procedures must be established.

The CDP-to-DDP migration is, at present, in its last stage. The current status of the system is as follows:

1. The finance and payroll applications have been distributed. The centralized data, mostly in the form of IMS databases, is accessable to users through interactive and batch applications.

2. The support system consists of a network of IBM 8100 processors interconnected in a star topology to the central IBM 3033 computer system. Each 8100 supports two types of terminals: a) local terminals connected through a loop to the 8100 processor and b) remote clusters of terminals connected to the 8100 processors through communication links. Each 8100 system has its own operating system (in our case DPCX), which allows: a) storage and access of local files, b) remote job entry workstation operations in conjunction with the host, c) direct access from 8100 terminals to the host IMS applications and d) several management and diagnostics tools. The update synchronization is performed by the users. The distributed communication system consists of IBM's SNA with fixed routing and centralized session control. At present, there is no distributed resource sharing.

3. The organizational system has been conditioned to handle the migration project. A detailed project plan has been prepared and followed rigorously. The plan includes executive sessions, equipment ordering, support system preparation, training, equipment installation, monitoring and project closing for each 8100 installation. The organizational system has also been oriented towards the formalization of problem/change management and a matrix-type project team has been at the core of the migration.

We encountered some hardware/software problems in the early phases of the migration which required detailed problem control and tracking. The support system problems have almost vanished. More work needs to be done in the area of applications as we move out of the migration task.

4. SUMMARY AND CONCLUSIONS

In this short paper, an attempt has been made to classify and analyze the problems facing a potential DDP installation. Possible approaches available to handle these problems have been discussed in general terms and then illustrated through an implementation example. To summarize:

1. Some applications are more suitable for distribution than others. The decision to distribute a given application needs to be carefully examined.

2. At present no suitable distributed database manager is available commercially. Thus update synchronization and distributed data models must be individually programmed. It is, however, expected that in the next four to five years several distributed database managers will be commercially available.

3. Well established hardware as well as software interfaces exist for DDP in homogeneous (same vendor) environments. Future improvements in the communication systems and network operating systems, combined with the standardization efforts of ANSI/ISO committees , will make interfaces between heterogeneous systems possible and economically feasible.

4. The conversion from CDP to DDP presents several technical as well as management problems. While the technical problems are numerous, the management problems are the most difficult to solve and need special attention.

ACKNOWLEDGEMENTS: The critique and editing by Warner Mach, my colleague and friend, is greatly appreciated.

BIBLIOGRAPHY

a). GENERAL INFORMATION

ADIB78a). Adiba, M. and Chupin, J.C; 'Issues in Distributed Data Base Management Systems: A Technical Overview', 4th interntl Conference on Very Large Data Bases, W. Berlin, Sept. 1978, pp.89-110.

ANDE75a). Anderson, G. & Jensen, E.D., 'Computer Interconnection Structures: Taxonomy, Characteristics & Examples', CACM, Dec. 1975.

ASCH74a). Aschim, F.,'Data Base Networks-An Overview', Management Information, Vol.5.1, Feb.1974, pp.12-28

BAGL79a). Bagley, J.D., 'Some Definitions Related to Distributed Data Processing', SHARE draft, 03/15/79

CHAM79a). Champine, G.A., 'Trends in Database Processor Architectures', IEEE Compcon, spring 1979, pp.69-71.

CHAN77a). Chandy, K.M. 'Models of Distributed Systems', IEEE pp.115-120

DAVI79a). Davies, D.W. et al, 'Computer Networks & Their Protocols', John Wiley & Sons, 1979.

DOLL78a). Doll, D.R.,'Data Communications- Facilities, Networks and Systems Design', John Wiley & Sons, N.Y. 1978

DOWN76a). Down, P.J and Taylor,E.F., 'Why Distributed Computing', NCC Publications, Manchester 1976

FRY76a). Fry,J. and Deppe, M., 'Distributed Data Bases- A Survey of Research', Computing Networks, Vol.1, No.2, 1976,pp.1-12

GANZ76a). Ganzhorn, K., 'Evolution in Communication Systems', Proc. of 6th Infmatics Symposium, Bard Hamburg, Nov. 1976, pp. 361-380.

KIMB75a). Kimbleton,S.R. and Schneider,G.M., 'Computer Communication Networks:

Approaches, Objectives & Performance', ACM Computing Surveys, Vol.7,No.3, Sept 1975

KLEI76a). Kleinrock,L.,'Queuing Systems', Vol.2, John Wiley & Sons , 1976

MARI79a). Mariani, M.P. and Palmer, D.P., 'Tutorial: Distributed System Design', IEEE Catalog No. EHO 151-1, 1979.

MCGL78a). Mcglynn,D,R., 'Distributed Processing & Data Communication', John Wiley & Sons, 1978

OKEE78a). O'Keefe, P.M.W., 'Distributed Processing and Corporate Committment', IEEE Compcon, Spring 1978, pp.348-350

PARG78a). Pargellis, A.M., Coffey, T.W., Moraski, R.M., 'In-Depth Assessment of the Status of Distributed Databases and Potential U.S. Navy Requirements', Data Solutions Corporations, Vienna, Virginia, june 1978.

PATR80a). Patrick, R.L., 'Application Design Handbook for Distributed Systems', CBI publishers, Boston, 1980.

ROME80a). Rome Air Development Center, 'Distributed Database Technology: State of the Art Review', RADC-TR-80-7,1980.

ROTH77a). Rothnie,J.B. and Goodman, N. 'A Survey of Research & Development in Distributed Data Base Management', 3rd Conference on Very Large Data Bases, Tokyo, Oct. 1977, pp.48-60

SALT78a). Saltzer, J.H., 'Research Problems of Decentralized Systems with Largely Autonomous Nodes', Operating Systems, Lecture Notes in Computer Sciences, 60, Edited by G.Goos and J.Hartman, Springer Verlarg, N.Y. 1978, pp.584-593.

TRIP80a). Tripathi, A.R. et al, 'An Overview of Research Directions in Distributed Processing', COMPCON fall 1980.

b). DDP SUPPORT SYSTEMS

ADIB78b). Adiba,M; Caleca,J.Y. and Euzet, 'A Distributed Database System Using Logical Relational Machines', 4th Interntl. Conf. on Very Large Data Bases, W. Germany, Sept. 1978, pp.450-461

BACH78b). Bachman,C., 'Distributed Reference Model', SHARE 51, August 1978, Boston, pp.154-160

BAKE77b). Baker,J. 'Distributed Processing: Modeling and Building Blocks', SHARE 49, Houston, Aug. 1977, pp. 65-70

BANE78b). Banerjee, J. et al, 'Concepts & Capabilities of a Database Computer', ACM Trans. on Database Systems, Dec.1978, Reprinted in IEEE tutorial on Centralized & Distrib. Databases

BAUM76b). Baum, R.I. & Hsau,D.K, 'Database Computers-A step towards Data Utilities', IEEE Tran. on Computers, Dec. 1976.

BERN78b). Bernstein, P.A. et al, 'The Concurrency Control Mechanism of SDD1: A system for Distributed Database', IEEE Tran. on Software Engg., vol. SE-4, NO. 3, May 1978.

CYPS78b). Cypser,R.J., 'Communications Architectures for Distributed Systems', Addison-Wesley, 1978.

DATE79b). Date, C.J.,'Locking & Recovery in a Shared Database System: An Application Programming Tutorial', VLDB Conf. Brazil 1979. pp.1-10

DAVI73b). Davies, C.T. ,'Recovery Semantics for a DB/DC System', ACM National Conf., 1973, pp.140-146.

DECN78b). DECNET, Digital Equipment Corporation Document, 1978.

DESJ80b). Des Jardinis, R.P, 'Reference Model for Distributed Systems', COMPCON, fall 1980.

DESJ78b). Des Jardinis, R.R. and White, G.; 'ANSI Reference Model For Distributed Systems', IEEE Compcon fall 78, Washington, pp. 144-159

HONE78b). 'Distributed Data Processing Technology', vol. 15W6, Honeywell Research Center, Minneapolis, 1978.

ELLI77b). Ellis, C.A., 'A Robust Algorithm for Updating Data in Duplicate Databases', Proc. 2nd Berkeley Workshop on Distributed Data Management & Computer Networks, 1977. Reprinted in IEEE tutorial on Centralized & Distributed Databases.

FELD79b). Feldman, J.A., 'High Level Programming Languages for Distributed Computing', CACM, june 1979, pp.353-370.

FORS78b). Forsdick, H.C. et al, 'Operating Sysems for Computer Networks', Computer, Jan 1978, Reprinted in IEEE tutorial on Distributed Processing.

GARC79b). Garcia-Molina, H. 'Performance of Update Synchronization Algorithms For Replicated Data in a Distributed Database', Ph.D. Dissertation, Stanford University, June 1979.

HOLG78b). Holmgren, J.F., 'Resource Sharing Unix', COMPCON, Fall 1978, pp.302-305.

IBM75b). IBM Manual, 'System Network Architecture-General Information' Manual no. Ga27-3102-0, 1975

IBM78b). IBM Manual, 'An Introduction to IBM 8100 System', Manual no.Ga27-2875-0, 1978

LIEN78b). Lien, Y.E.,& J.H. Yong, 'Design of a Distributed Entity-Relationship Database System', COMPSAC 1978, Reprinted in IEEE Tutorial on Centralized & Distributed Databases.

LORI80b). Lorin, H., 'Key Decisions in Distributed Processing' , COMPCON, Fall 1980.

ROTH77b). Rothnie, J.B. and Goodman,N., 'An Overview of Preliminary Design of SDD1: A System For Distributed Data Bases', Proc. of 2nd Berkeley Workshop on Distributed Data Management & Computing Networks, May 1977, pp.39-57

SILB80b). Silberchatz, A., 'A Survey Note on Programming Languages For Distributed Computing', COMPCON Fall 1980, pp.719-723

SMIT80b2). Smith, L.W, 'Distributed Systems Management: Key Issues', COMPCON fall 1980.

STON75b). Stonebroker,M. and Neuhold,E., 'A Distributed Database Version of INGRES', Proc. of 2nd Berkeley Workshop on Distributed Data Management & Computing Networks, Manual, Ga27-3102-0, 1975

THOM78b). Thomas,R.H., 'A Solution to the Concurrency Problem for Multiple Databases', IEEE COMPCON, Spring 1978, pp.56-63

THUR80b). Thurber, K.J., 'An Assessment of the State of Network Architectures', COMCON fall, 1980.

WECK76b). Wecker,S. and Conant,G. 'DNA: An Architecture for Heterogeneous Computer Networks', Proc. of 3rd International Conference on Computer Communications, Toronto, Aug.1976

c). DISTRIBUTED APPLICATION SYSTEMS

BUCC79c). Bucci,G., Streeter,D.N., 'A Methodology for the Design of Distributed Information Systems', CACM,v.22, no.4, April 1979, pp. 233-245.

BUCK79c). Buckles, B.P. and Hardin, D.M., 'Partitioning and Allocation of Logical Resources in a Distributed Computing Environment', General Research Corporation Report, Huntsville, Alabama, 1979.

CASE72c). Casey, R.G., 'Allocation of Copies of a File in an Information Network', SJCC 1972, AFIPS Press, Vol. 40, 1972

CASE73c). Casey, R.G., 'Design of Tree Networks for Distributed Data', AFIPS Conf. Proceedings, Vol. 42, 1973, pp.251-257.

CHAN76c). Chandy, D.M. and J.E. Hewes, 'File Allocation in Distributed Systems', Proc. of the Intl Symp on Computer Performance Modelling, Measurement and Evaluation, March 1976, pp.10-13.

CHAN76c). Chang,S.K.,'A Model of Distributed Computer System Design', IEEE Tran. on Systems, Man, and Cybernetics, Vol.5, No.6, May 1976, pp. 344-359.

CHU69c). Chu, W.W., 'Optimal file allocation in a multiple computer system', IEEE Tran. on Computers, oct. 1969, pp. 885-889.

CHU79c). Chu, W.W., and Hurley, P., 'A Model for Optimal Query Processing for Distributed Databases', IEEE Compcon, Spring 1979, pp.116-122.

COFF80c). Coffman, E.G., Gelenbe, E, et al, 'Optimization of the Number of Copies in Distributed Databases', Proc. of the 7th IFIP Symposium on Computer Performance Modelling, Measurement & Evaluation , May 1980, pp. 257-263.

DINK78c). Dinkel, J.J. and Kochenberger, G.A, 'Resource Allocation in Computer Networks by a Regional Access Model', Computers and Operations Research, Vol. 5, pp.139-147, 1978.

ESWA74c). Eswaran, K.P; 'Allocation of Records to Files and Files to Computer Networks', IFIP, 1974, pp.304-307.

GYLY76c). Gylys, V.B. and Edwards, J.A, 'Optimal Partitioning of Workload for Distributed Systems', COMPCON, fall 1976.

IRANI81c). Irani, K.B. and Khabbaz, N.G., 'A Combined Communication Network Design and File Allocation for Distributed Databases', 2nd Intl Conf on Distributed Systems, Paris, April 1981.

KHAB80c). Khabbaz, G.N, 'A Combined Network Design and File allocation in Distributed Computer Networks', Ph.D dissertation proposal, Univ. of Michigan, 1979.

KIMB77c). Kimbleton,S.R; 'A Fast Approach To Network Data Assignment', Proc. of 2nd Berkeley Workshop on Distributed Data Management & Computing Networks, May 1977, pp.245-255

LEVIN75c). Levin,K.D; and Morgan,L.H; 'Optimizing Distributed Data Bases-A Framework For Research', Proc. NCC, 1975, Vol.44, pp.473-478

LOOM76c). Loomis,M.S; and Popack,G.J; 'A Model for Database Distribution', IEEE Computing Network Symposium, Gaithusburg, 1976, pp.162-170

MAHM76c). Mahmood, S. and Riordan, J.S.; 'Optimal Allocation of Resources in Distributed Information Networks', ACM Trans. on Database systems, Vol.1, No. 1, March 1976, pp. 66-78

MARC81c). Marcogliese, R. and Novarese, R., 'Module and Data Allocation Methods in Distributed Systems', 2nd Intl Conf on Distributed Computing Systems, Paris, April 1981.

MORG77c). Morgan, H.L; and Levin, K.D; 'Optimal Program and Data Locations in Computer Networks, CACM, Vol. 20, No.5, May 1977, pp.345-353

SURI79c). Suri, Rajan, 'A Decentralized Approach to Optimal File Allocation in Computer Networks', Ph.D. dissertation, MIT, 1979.

TABA78c). Tabatabi, V. and Arizoulla, M., 'A Queue Theoretic Model for File Assignment in a Distributed Data Base Network', IEEE Compcon, 78, pp.175-183.

UMAR81c). Umar, A., 'Allocation of Programs/Databases in Distributed Data Processing Environments', Dissertation Proposal, IOE Department, Univ. of Mich, 1981.

WHIT70c). Whitney,K.M; 'A Study of Optimal File Assignments and Communication Network Configurations', Ph.D. dissertation, U. of Michigan, 1970

d). CONVENTIONAL APPLICATION SYSTEMS

ALFO79d). Alford, M.W., 'Software Requirements Engineering Methodology (SREM) at the age of Two', COMPSAC 78. Reprinted in Tutorial on Distributed Systems Design . IEEE Catalog No. EHO 151-1, 1979.

ARON69d). Aron,J.D.,'Estimating Resources for Large Scale Programming systems', NATO Science Conference, Rome Oct. 1969

BELA76d). Belady and Lehman, 'A Model of Large Program Development', IBM Sys. Journal, vol. 15,no.3,1976.

BELA77d). Belady and Lehman, 'Large Program Development', Conference on Reliable Software, April 1977.

BOEH73d). Boehm, B.W.,et all, 'Characteristics of Software Quality' ,TRW Systems Group, Report No. 25201, 6001-R0-00, Dec.1973.

BOEH74d). Boehm, B.W., 'Cost of Software', paper published in in Practical Strategies for Developing Large Scale Software, edited by Horowitz, Edison Wesley.1974.

CHEN77d2). Chen, P.P., 'The Entity-Relationship Model - A Basis for the Enterprise View of Data', AFIP Conf., Vol. 46, 1977 NCC

CHEN76d). Chen, P.P., 'The Entity-Relationship Model - Towards A Unified View of Data', Trans. on Database Systems, Vol. 1, No. 1, March 1976

DALY77d). Daly, E.B, 'Management of Software Development', IEEE Trans. on Software Engg., May 1977.

FRY78d). Fry,J.P; and Teorey,T.J; ' Design & Performance Tools for Improving Database Usability and Responsiveness', Univ. of Michigan, Data Base Research Group Technical Report 78DB10, March 1978

GANS76d). Gansler,J.G, ' Keynote: Software Management', Symposium on Software Engg., April 1976.

KEND77d). Kendal,R.c.,' Management Perspectives on Programs, Programming and Productivity', GUIDE Conference, Oct. 1977.

PARNA72d). Parnas, D.L, 'On the Criteria to be used in decomposing Systems into modules', CACM, Dec. 1972.

PARNA72d2). Parnas, D.L, 'A recent technique for software module specification with examples', CACM, May 1972.

PRES75d). Presser,Leon, 'Software Management', ACM annual Conference,Oct.1975

PUTN78d). Putnam, L.H., 'A General Empirical Solution to the macro software sizing and Estimating Problem', IEEE Tran. on Software Engg., Vol. SE-4,No.4, July 1978, pp.345-360.

PUTN76d). Putnam, L.H.,'A Macro Estimation Methodology for Software Development', COMPCON, Fall 1976, pp.138-143.

SENK77d). Senko, M.E.,' Data Structures and Data Accessing in Data Base Systems past, present, future', IBM sys. Journ. Vol. 16, No. 3 1977, pp. 208-257.

TEICH74d). Teichroew, D., 'Improvements in System Life Cycle', IFIP Proceedings, 1974, 972-978.

TEICH77d). Teichroew,D., 'Computer Aided Software Development', Conf. on Reliable Software, April 1977.

TEICH70d). Teichroew, D., 'Problem Statement Languages in MIS', Proc. sysmposium on BIFOA, 'Management Information Systems- A challenge to Scientific Research', Cologne, July, 1970, pp.253-270.

TEOR80d). Teorey, T.J. & Fry, J.P., 'The Logical Record Access Approach to Database Design', ACM surveys, Fall 1980.

THAY75d). Thayer, R.H.,'The Air Force Software Manager', ACM Annual conf. Oct. 1975.

TSCH78d). Tschritzis, D. and Klug, A., 'The ANSI/X3/SPARC DBMS Framework', SHARE 51, Boston, Aug 1978, pp.320-332.

WALS77d). Waltson & Felix,'A Method of programming measurement and estimation', IBM System Journal, Vol. 16, No. 1, 1977.

CRITERIA FOR ESTABLISHING A COMPREHENSIVE COMPUTER PROGRAM FOR A SMALL LIBERAL ARTS COLLEGE

Jane H. Wilde

Director of Curriculum Development
Voorhees College, Denmark, S. C. 29042

American society is being transformed by a quiet revolution which involves the information explosion and the increasing use of the computer. However, higher education seems, to a significant extent, unmoved by these changes. Activity that has taken place has focused on the need for establishing majors in computer science and on teaching the use of the computer in areas where it is necessary to manipulate large amounts of numerical data. Arguments for establishing programs based on these considerations are unquestionably valid, but they do not go far enough. In today's world, it is vitally important that all students graduating with a liberal arts degree possess basic computer literacy. Ability to utilize the computer may soon become one of the basic skills. Increasing numbers of students are beginning to enter college with at least two years of previous experience with the use of the computer. It is clear that students who have been exposed to the computer will never learn in the same way. The c-mputer compels the development of heightened skills in precision, organization, logic, operator responsibility, and clarity. The economic pressures of the present day and new technologies should force colleges which want to remain viable to respond quickly to the needs of providing a basic computer literacy for all graduates. Any college which has as its mission enabling its graduates to be fully prepared to confront life in a rapidly changing world can do no less.

Evidence clearly shows that the use of educational computing is not spread equally among the 50 states or among all social groups. Studies show that persons living in the southeast, women, minorities, and persons living in rural areas are the least likely to have any experience with the computer. The following figures illustrate the problem for minority institutions. About 70% of all institutions (including minority institutions) reported they had computing equipment, tending to suggest all is well. The critical fact, however, is that only 3% (or one-fortieth) of the total number of students having experience with computers were in minority colleges.

Role of Computer in Educational Program

How does a small liberal arts college in the southeast confront this challenge to develop a comprehensive program? Educational aims, funding, personnel, hardware and software decisions must be made with a full realization of the implications for social and economic equality, productivity, and the well being of the nation. Experience shows that that as educational computing programs are developed, initially there was a natural tendency for the computer program to develop in the mathematics and natural science departments and for the computer programming courses to be developed as a part of the mathematics curriculum, or with those courses that involve statistics and data processing. Sources for funding also pushed the college in this direction as the initial impetus came from an effort to develop a grant proposal to increase minority participation in mathematics. At the same time, some faculty were concerned about developing a comprehensive program in computer-assisted instruction, starting with the basic skills courses. As the discussion broadened, it was noted that identifying computer science with a program in mathematics was not necessary nor desirable and could create a computer anxiety related to math anxiety, producing unnecessary roadblocks to the development of computer literacy. As a result a campus wide committee came into being, including all departments, to develop a comprehensive computing program, once the need for such a program was seen as the desired objective. It became apparent that the special objectives of a mathematics program and the goals of a comprehensive instructional computing program were inseparable. In fact, the use of the computer in CAI and its widespread use in the basic skills program would create familiarity with its use in addition to providing a patient task master for drill and practice. It would have the potential of creating a student interest in computers where none had existed before and might also lead students to consider computer science as a potential major. It also had the additional advantage of spreading the cost over the entire student body rather than maintaining expensive equipment for the almost exclusive use of a relatively few majors. It was also agreed that the over-all goal of educational computing on the campus was to provide computer literacy for all students. Once this broad purpose was agreed upon, it was natural to state that the computer science major and the use of CAI should be available to all areas, not just mathematics.

There are strong arguments for the use of CAI throughout the curriculum, starting with the basic skills. CAI does provide for truly individualized instruction and is a cost effective way of handling drill and practice. Far from replacing the teacher, the use of CAI will free up valuable faculty time, enabling instructors to work with the student in tutorial sessions. Students who enter college and are placed in the basic skills program will be given an introduction to the use of the computer and instruction in entering responses in the programs. Students who test out of the basic skills will also receive this same training. All students will be required to take a broadly based course, Introduction to Computer Science, which will discuss the role of the computer, its uses, and will teach them to write a very simple program. Areas in which CAI will be used, beyond the basic skills courses, particularly mathematics, courses in data processing, accounting, journalism, sociology, political science, to name but a few. CAI will also be useful in helping students prepare for such tests as the NTE, GRE, and MCAT.

An auxiliary problem is that of training faculty to teach in a CAI program. This is not insuperable. There are numerous examples of faculties that have become adept in developing their own instructional programs, using an author language. Bennett College in Greensboro, North Carolina, is an excellent example of this. Faculty were familiarized with the computer and the terminals during inservice workshops. They were given grants in summer to develop programs with payment tied to the completion of the agreed upon tasks. Probably the best way to overcome faculty computer anxiety is to make it possible for them to have some hands-on experience in entering responses in programs and in developing very simple programs of their own so that they can discover that they will not break the computer. Any instructorwho has had some training in the instructional use of computing will never teach the same way again and will be better able to teach students now entering college with prior experience with computers.

It is perhaps easier to agree on broad principles than to solve the many practical problems of implementation. Decisions must be made concerning hardware, personnel and, how computer science is to relate to the majors.

Structure of the Major

After much consideration, the decision was to develop a flexible computer science major with a core of computer science, and programming courses. This proposal is for a computer science major with emphasis in accounting, --- with emphasis in Social Sciences, --- with emphasis in mathematics, in Business Administration, in Biology, in Education. The proposed major will include 6 hours of statistics, 3 hours of data processing, 12 hours of computer science, 9 hours of computer programming, and 12 hours in area of emphasis courses. It is a structure that

90

will accomplish its objective with a minimal addition to course offerings.

Hardware Considerations

To implement a major of this sort along with the widespread use of computer assisted instruction will require at a minimum, a mini-computer with some peripherals including card-processing equipment and terminals. Another must is the capability for using an author language. Although micro-computers would be adequate for CAI in some areas, the great advantage of the mini-computer is that the computer can keep track of student use and student progress. It can help the student and the instructor to keep track of what materials the student has completed satisfactorily. It will enable the student to plug into the program at the point where he left it without having to start again at the beginning. The micros are also more easily transportable and there is more danger of pilferage. The program as it develops, would not rule out micros, but the major thrust is toward the acquisition of the mini.

Other questions were addressed as decisions were made. Where is the computer to be housed and what building renovations need to be made? What changes need to be made in power supply? What needs to be done to insure proper environmental control? As hardware is identified which will best serve the identified purposes, it is important to make certain that timely maintenance service can be obtained and that adequate service agreements can be obtained which will assure next-day repairs when necessary.

Personnel

Without question, additional personnel will be required. A faculty member with at least a master's degree in computer science will be added to serve as the Director of the Educational Computer Program. A person holding the degree is needed not only to meet standards for accreditation of the major, but also to provide advice during installation, to teach courses, and to supervise the day-to-day operation of the computer. The program would be strengthened by adding a data entry operator to assist the computer director and to assist the faculty with the development of CAI programs.

The problem of personnel is a knotty and serious problem. To employ a data entry operator will be easy compared to the problem of attracting and employing a computer director. In our experience, as well as in larger, prestigious institutions, it is virtually impossible to attract the qualified person. Obviously the computer director will need some notion of instructional design or must be able to work closely and effectively with faculty to orient them to the development and use of CAI programs. Cost is the primary obstacle. The scarcity of persons coupled with the demand from industry and business has driven the salaries of persons with only the minimum qualifications far above the salaries commanded by full professors and chairpersons at many small institutions.

One solution to the problem is for the college to train its own personnel. (It is especially advantageous if the person also has a genuine committment to the college.) Programs are often available with special funding which will enable a college to have one of its faculty members fully qualified at the end of three summers. The NSF has also provided funding for a variety of educational computing programs.

(Another question related to personnel and the accreditation of liberal arts colleges is the requirement that each major have at least one doctorate in the major area. The extreme difficulty of hiring such a person poses a real problem from the small college.)

Concluding Observations

In spite of the substantial problems which need to be overcome for the development of an educational computing program, colleges really no longer have the choice. Both teaching and learning are being influenced and changed by the computer and the information explosion. Is a student graduating with an accounting major or seeking a career in journalism really adequately prepared in today's world without the ability to access and use computers? For colleges to remain unresponsive is to insure their obsolescence.

Is is also essential that broad national policies be established regarding computer literacy. The already existing imbalance in access to educational computing facilities needs to be addressed. (Recall that the southeast, minorities, women, persons in rural areas lag behind.) If the nation remains committed to opportunity for all, consciousness needs to be raised among the public, educators, and policy makers.

The lack of national policy and the failure of educational institutions with respect to educational computing will leave the disadvantaged even more disadvantaged. To regard computers as an unnecessary frill in higher education is folly. The need is to develop an instructional system that will equip students to adapt themselves to changing situations and to utilize the computer to enlarge their base of knowledge which will leave the college on the cutting edge of the future. The task is difficult, but the need is urgent.

BIBLIOGRAPHY

Johnson, D. C. , Anderson, R. E. Hausen, T. P., Klassen, D. L. "Computer Literacy - What Is It?" Mathematics Teacher, 73(February, 1980), 91-96.

Diem, Richard A. "Developing Computer Skills: An Inservice Training Program". Educational Technology, Fedruary, 1981, 30-32.

Johnson, James W. Getting from Here To There: The Status of Instructional Computing in Higher Education, T.H.E. Journal, 7(November, 1980), 48-53.

Lautsch, John C. The Computer and Education: The Genie is Out of the Bottle, T.H.E. Journal. 8(February, 1981), 34-35, 39.

Leuerman, Arthur, Computer Literacy, EDUCOM, Winter, 1981.

Marshall, Sister Patricia. Instructional Computing in Minority Institutions, A Needs/Strategy Assessment. Executive Summary of Final Report/NSF SPI 7821515, 1980.

Molnar, Andrew R. The Next Great Crisis in American Education: Computer Literacy. Journal of Educational Technology Systems, 7(1978-79), 275-284.

Shostak, Arthur B. The Coming Systems Break: Technology and Schools of the Future. Phi Delta Kappan, 62 (January, 1981), 356-359.

Wilde, Jane H. The Computer in Developmental Education, Proceedings, AEDS, May 1981, 290-294.

____________ Computers May Help Basic Skills Students. Electronic Education, September, 1981, 27-28.

SUCCESSFUL IMPLEMENTATION OF COMPUTING TECHNOLOGY IN SCHOOLS THROUGH TEACHER EDUCATION, PROFESSIONAL DEVELOPMENT AND SUPPORT

1982 May 11

By Cam Willis
Project Officer
Computer Assisted Learning
Board of Education for the City of London
165 Elmwood Avenue
LONDON, Ontario Canada

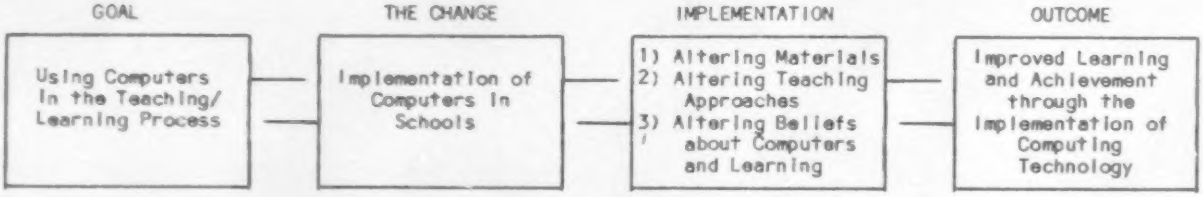
Successful implementation of new ideas, new curricula or new teaching styles is a very complex and difficult issue. Planning and coordinating implementation of new curricula often results in failure because those charged with implementation fail to address the dynamic social process inherent in any implementation scheme or model.

This paper is an attempt to create a plan that can be utilized by a school, or school system as it implements computing technology into any classroom at any grade or subject level. It will not address the financial or physical characteristics inherent in the change but will discuss how one can successfully manage the human side of this enterprise through teacher education, support and professional development.

Specifically, it will address the following issues:

1. What is implementation?
2. What factors affect it?
3. What can school systems do to make implementation a positive and successful event?
4. What is the role of teacher education professional development and support in the implementation of computing technology?
5. How is it done in London, Canada?

THE LOGIC OF IMPLEMENTATION OF COMPUTING TECHNOLOGY INTO THE INSTRUCTIONAL PROCESS


GOAL
THE CHANGE
IMPLEMENTATION
OUTCOME
Using Computers in the Teaching/ Learning Process
Implementation of Computers in Schools
1) Altering Materials
2) Altering Teaching Approaches
3) Altering Beliefs about Computers and Learning
Improved Learning and Achievement through the Implementation of Computing Technology

Fullan & Park (1980)

> To many educators, implementation is simply a change, a revision, an innovation. Implementation is far more complex than any of these. Implementation of any new curriculum or teaching strategy consists of alterations from existing practice to some new or revised practice (potentially involving materials, teaching approaches and beliefs) in order to achieve certain desired student learning outcomes.
>
> (Fullan & Park 1980)

All implementation plans spring from a desire to reach a new or different goal than is currently being pursued. In our school system, we believe that computers can have a very positive effect upon learning outcomes and a key system goal is the successful implementation of that technology into all our schools and classes.

The critical area in the Logic of Implementation lies in the implementation plan. All three factors, new materials to be used in computer-aided learning, new approaches and strategies to be tried, and a positive valuing by educators regarding computers and instruction must be carefully weighed if the plan is to be successful. The people resources are paramount and must be strongly considered in planning. It also should be stated that the flow is not always linear from the goal to the outcome stage. It may in fact be just the reverse. As strategies are tried, and experiences become reinforced, there will be an ebb and flow in both directions. This ebb and flow can be to the benefit or detriment of the implementation scheme.

The implementation of any new curricula will be difficult because these are difficult times for public education. Declining enrolments, high rates of inflation, high unemployment in many economic sectors have caused serious strains on the public school system. Coupled with an increasingly jaundiced view of public schools by the taxpayer, it is no wonder that many school systems' implementation plans fail.

These factors are external to the real issue, over which educators have little control. Because educators are faced with the consequences of factors like inflation, declining enrolment and a skeptical public, the implementation plan should take those externalities as given and proceed from that juncture.

There are twelve factors that we can identify and plan for in our implementation scheme. Each of these factors must be understood and addressed by the planners charged with the implementation process.

Two significant paradigms govern all twelve factors. The first is that all twelve factors will operate in a dynamic, not static, mode because implementation is a people-oriented process. The second paradigm is that all factors form a system of interactive variables that will become both positive and negative influences to the implementation process throughout the life of the plan.

THE TWELVE FACTORS AFFECTING IMPLEMENTATION

A) Characteristics of the Innovation or Revision

1) Need for change
2) Clarity, complexity of the change
3) Quality and availability of materials

B) Characteristics at the School System Level

1) Previous history of innovation attempts
2) Expectations and training for principals
3) Teacher input and professional development
4) Board and community support
5) Time line and monitoring
6) Overload

C) Characteristics at the School Level

1) Principal's actions
2) Teacher/teacher relations and actions

D) Factors External to the School System

1) Role of the Ministry of Education and other educational agencies

(Fullan & Park 1980)

The implementation plan must address each of these factors. Many schools and school systems create a tailored item-specific curricula that meets their system's needs. What is missing, however, is the process through which materials, teaching approaches and teaching beliefs can develop over time. Without a strong emphasis upon process development, implementation will be limited.

The socialization process, so necessary in curriculum implementation planning, will only occur if the implementation plan makes provision for it. It is the process and resocialization involving all partners in the change process that makes implementation happen.

Ten principles underlie any implementation process. Our computer implementation plan and all other new curricula will be governed by these principles.

PRINCIPLES OF CURRICULUM IMPLEMENTATION

1) Implementation is a dynamic process, not a static event.

2) The innovations will get adopted, further developed and modifed over time.

3) Implementation is a process of professional development and growth which must be supported and nurtured.

4) Implementation is a process of clarification.

5) Because implementation is a socialization and clarification process, interaction and technical assistance are essential.

6) Planning at the system and school level is a necessity if obstacles are to be addressed.

7) Plans to guide action must:
    a) be based on these principles;
    b) address outcomes (materials, approaches, beliefs) systematically;
    c) coordinate strategies to address the twelve factors;
    d) monitor the progress in order to assess progress and solve problems.

8) Developing and using an implementation plan is in itself an implementation problem.

9) One hundred percent implementation is probably not desirable and in any case is impossible to achieve.

10) The ultimate goal of implementation is not to implement a particular innovation, but to develop the "capacity" for systems, schools and individuals to process for innovation, change and revision.

(Fullan & Park 1980)

THE LONDON PLAN FOR SUPPORT, EDUCATION AND DEVELOPMENT

Computer Assisted Learning Department

Laboratory One

Laboratory Two

Teacher Education

1) 5 hour course
2) 12 hour course
3) 24 hour administrator's course
4) 3 part (80 hour term) spcialist's course
5) specific courses

Teacher Support/System Support

1) software evaluation
2) software modification
3) software authoring projects
4) specific curriculum projects

Community Support

1) PTA/Home & School, Service Club assistance
2) community groups, churches, professional organizations

Our needs assessment begun in 1979 indicated that successful implementation of computing technology into our instructional areas would require strong and on-going support. Studies, conferences and visitations to other jurisdictions indicated that new technology, like computers, creates anxiety in people, especially teachers, and that teachers must view their expenditure of time and energy spent upon professional development as productive if they are to inculcate their efforts and experiences into positive beliefs about computers and instruction.

Our plan involves a major commitment on our Board's part over the next decade. In 1980, a new department, Computer Assisted Learning, was created to plan, coordinate, acquire and successfully implement computing technology and applications in all jurisdictions but Business Education, Data Processing. In 1981, an instructional laboratory environment was created solely for support and professional development purposes. Another is currently under construction for September 1982.

The purposes of these laboratories are as follows:

1) To provide professional development, support and staff education programs for all our employees.

2) To acquire, modify or create good courseware that will meet our system's needs, goals and objectives.

3) To provide "hands-on" computing experiences for all grade five students taking the computer awareness unit.

STAFF EDUCATION, SUPPORT AND PROFESSIONAL DEVELOPMENT

A review of the ten principles governing implementation indicates that the extent and depth of change is dependent upon people. What they will or will not do determines success or failure. All our support in this area is a concious attempt to help people change. The more positive the impact, the more likelihood of success in this venture.

Each laboratory facility is open six days a week. During that time specific activities take place in the laboratory environment.

I TEACHER EDUCATION

A) Computers in the Classroom

This four-hour session instructs teachers what a computer is and what it can and cannot do. The purpose of this unit is to acquaint teachers with computing technology, defuse anxiety through a positive experience with it, and illustrate how it can be used in their classroom effectively.

B) Computer Awareness

This twelve-hour (two Saturday sessions) course is an intensive course on using and communicating with a computer. The key to this course's success is that every participant takes the microcomputer, related software and assignments away with them at the end of the day. They have at their disposal both at home and at school, a computer for two full weeks. It gives the teachers a good opportunity to work with it at their pace in their own home and to observe children working with a computer. Monitoring of this program indicates that many of the teachers are incorporating microcomputing technology into their belief system. This would be much harder to do if they did not have the computer at their disposal for the full course session.

C) Computers and Administration

Principals, vice-principals and school administrators are key implementation agents. Their active, supportive participation and leadership is vital because they are the "gate-keepers"; the facilitators in the implementation process. In Ontario, most principals have been trained in the administrative role very effectively. In curriculum implementation, however, they have received little, if any, training.

To help them gain an understanding and become facilitators in our implementation plan, a twenty-four hour, eight session course has been created. It has equal emphasis upon instruction and administration. At the end of these sessions, the principals, vice-principals and senior administration have much greater clarity with regard to the role computers will play in the instructional and administrative areas.

D) Computers and Education

Our needs assessment indicated that schools and school systems will need support personnel who can assist and support their colleagues in instructional computing applications. In conjunction with the FACULTY OF EDUCATION - UNIVERSITY OF WESTERN ONTARIO, we've created a three session, eighty hours per session, course called "Computers and Education". It is not computer science or data processing, but is a course designed to assist all educators in instructional computing applications.

E) Specific Courses

As needs arise, our department designs and presents courses or professional development activities that meet those needs. Because of the laboratory environment, our system can create activities that are relevant, personal and supportive.

II TEACHER SUPPORT/SYSTEM SUPPORT

Support for classroom practitioners involves both professional development and technical support if implementation is to be successful.

The Computer Assisted Learning Department helps all schools in our system acquire hardware and software. The department also modifies both hardware and software to meet specific needs for exceptional children.

A) Software Authoring Projects

Acquiring courseware that is educationally sound and meets our system's needs is very difficult to do. A major project funded by the federal government will create some two hundred and twenty-five programs this summer. A needs assessment indicated our major priorities should be placed in the language arts, science and social studies disciplines in the elementary school. Following Isaacson's model (Isaacson 1981), we shall attempt to alleviate that need by writing good instructional software that meets our needs.

B) Specific Curriculum Projects

The Computer Assisted Learning Department has facilitated two major curriculum units this past year. Both have been an attempt to create a curriculum unit plus an implementation plan that was dynamic, and people-oriented. It is the process, not the change that will make our two new units, GRADE FIVE COMPUTER AWARENESS and GRADE ELEVEN COMPUTER SCIENCE successful.

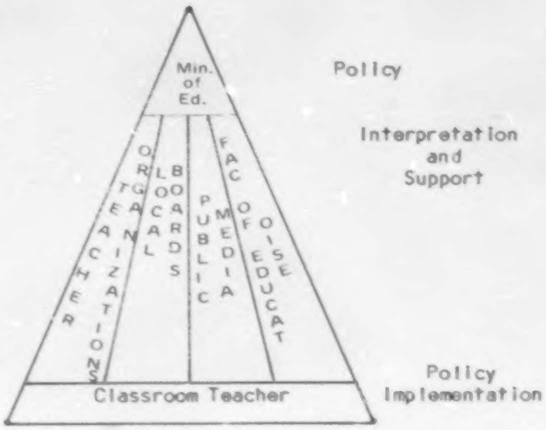
C) Community Support

Public education in my community, unlike many communities has an excellent reputation. Our school system has been recognized nationally as a leader in exceptional education, program development and program support services. Specific sessions for PTAs, Home and School groups and Service Clubs are given in our laboratory facility. Major addresses on computers and learning have been given by this department in order to gain broader understanding and to garner support for our efforts in our local, provincial, national, and international community.

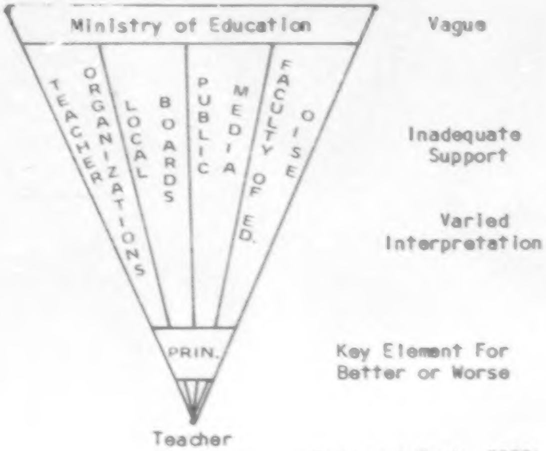
In the long run, our efforts, development and support are geared towards using the computer as more than an aid to be used in traditional drill and practice, simulation, or tutorial activities. We would hope, as Henry Olds stated "to use it to expand the intellectual capabilities of learners, to become an intellectual amplifier". Only when "the recognition that this technology is more significant to the growth of human conciousness than any other development" will public education move into real student-centred learning.

(Olds, 1981)

Throughout all of this implementation process, the teacher must view the change as intended.



Too frequently, implementation resembles this



(Fullan & Park 1980)

We believe that the plan we've created will generate successful implementation of computing technology into our schools' curriculum.

BIBLIOGRAPHY

Boyle, Harry J. Rights, Rhythms and Minerva's Owl. An Address given at 1981 Annual Ontario Association for Curriculum Development, Toronto, 1981.

Fullan, Michael and Park, Paul. Curriculum Implementation. Toronto: Ontario Ministry of Education, 1981.

Isaacson, Dan. How to Design Educational Microcomputer Programs. Microcomputers in Education: Getting Started. Proceedings of the Ninth Annual Math/Science Conference 1981. Arizona State University, Tempe Arizona; pp. 2.32-2.42.

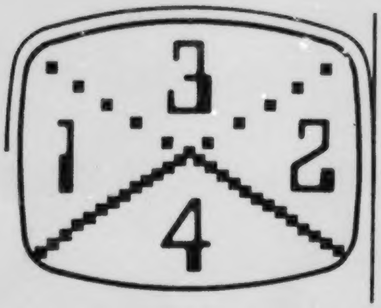
Morsund, David. Educating the Educators. Microcomputing, September, 1981 pp. 56-58.

Olds, Henry F. Through a New Looking Glass. Microcomputing September, 1981, pp. 62-74.

Poirier, Maurice & Garneau E.S. Goals for Educational Computing. Proceedings of the Educational Working Conference on an Instructional Computing Policy for Ontario. Ontario Institute for Studies in Education, Toronto, 1980 pp. 87-93.

EDUCATIONAL COMPUTING: SUPPORT FUNCTIONS AND STUDENT OBJECTIVES

DENNIS L. BYBEE, Ph.D.
COMPUTER EDUCATION COORDINATOR
DEPARTMENT OF DEFENSE DEPENDENTS SCHOOLS



Educational computing programs of the Department of Defense Dependents Schools can be described in terms of school level SUPPORT FUNCTIONS and STUDENT OBJECTIVES.

Student objectives include normal expectancies for student achievement in computer literacy and computer science. Computer literacy is defined to include a broad range of activities at a beginning level that are implemented through the math curriculum. Computer science objectives deal with more specialized topics such as programming, design and operation of computer hardware, and the use of computer systems in problem solving.

Student Objectives

All objective statements should be preceded by the phrase "THE STUDENT WILL BE ABLE TO. . . ."

1 Computer Literacy

1.1 Interact with a computer and/or other electronic devices.

1.2 Explain the functions and uses of a computer system.

1.3 Utilize systematic processes in problem solving.

1.4 Appraise the impact of computer technology upon human life.

2 Computer Science

2.1 Write structured and documented computer software.

2.2 Demonstrate knowledge of the design and operation of computer hardware.

2.3 Use computer systems in problem solving.

Support functions include both computer education services and school administrative support. Computer education services are defined as those functions that are in support of the entire school computer education program and include implementation of computer-based instruction in all of the instructional areas. School administrative support functions include student services, assistance to other education support personnel, support of automated management of school resources, and direct support of program planning, evaluation, and in-service activities in the school.

Support Functions

All functional statements should be preceded by the phrase "THE SCHOOL COMPUTER COORDINATOR SHOULD . . . ."

3 Computer Education Services

3.1 Assess school classroom instructional needs.

3.2 Advise school personnel of available hardware, software, and supplies which will support individual classroom needs.

3.3 Provide technical assistance through in-service and consultation.

3.4 Act as a point of contact for school computer activities.

4 School Administrative Support

4.1 Coordinate with school personnel in accomplishing computer-assisted STUDENT SERVICES tasks.

4.2 Assist EDUCATION SUPPORT PERSONNEL in the utilization of computer technology to accomplish their program support functions.

4.3 Provide assistance in support of the automated MANAGEMENT OF SCHOOL RESOURCES.

4.4 Support program PLANNING, EVALUATION, and IN-SERVICE activities in the school.

These definitions of support functions and student objectives are intended to identify and explain current and potential uses of computer technology, as well as to stimulate and provide direction for further development of computer utilization among elementary and secondary schools throughout the geographically dispersed dependents' schools system overseas. It is invisioned that full implementation of these objectives and functions will be accomplished over the next 3 to 5 years.

If the reader has any questions about these support functions and student objectives or would like more information about the Department of Defense educational computing program he/she should contact the author at the following address:

Department of Defense Dependents Schools
2461 Eisenhower Avenue
Hoffman Building #1, Room 148
Alexandria, VA 22331
(202) 325-7810

FLORIDA CENTER FOR INSTRUCTIONAL COMPUTING

Dr. C. William Engel
Director, FCIC

Introduction:

The Florida Center for Instructional Computing (FCIC) is located in the College of Education at the University of South Florida's main campus in Tampa. The Center is funded in part by the Florida Department of Education.

The major goal of FCIC is to support the use of instructional computing in Florida schools. The Center encourages the use of a variety of computers from micros to mainframes at all grade levels for the purpose of managing, supplementing and delivering instruction.

Need:

The College of Education at the University of South Florida has been involved in instructional computing for the past seven years. One of the first Altair microcomputer kits was construced and used by faculty and students in the College. A Heathkit microcomputer system was also constructed by students and some Radio Shack computers were purchased. The small microcomputer laboratory served undergraduate, graduate and inservice courses.

The need for an organized center to support instructional computing within the State became evident when requests for services far exceeded the ability of the College to meet the demands. School systems requested inservice activities, the College needed computer literacy courses and graduate programs, and the Continuing Education Department wanted non-credit courses. The College was already involved in a variety of activities related to instructional computing; however, the faculty could not retrain fast enough to meet the demand.

Centralization:

There are several advantages in establishing one center for instructional computing within the State. The success of the Minnesota Educational Computing Consortium (MECC) supports the need for centralization. Certainly the DOE, universities, school systems, professional groups and commercial vendors will provide a variety of services to users. However, there is a need for one focal point to coordinate the resources within the State and to communicate ideas related to instructional computing. A center with national visibility can work closely with commercial vendors and publishers to review and evaluate new products and to disseminate the information to users. One center can afford to experiment with new ideas and products. A team of experts involving a variety of colleges and departments within the university can address complex problems in instructional computing that would not be cost effective or even possible for a school system to address on an individual basis. Students involved in graduate programs in "Computers in Education" within the College of Education at USF are encouraged to support FCIC activities. The administration within the University and the College have encouraged the use of instructional computing activities. Space, staff, and financial support have been allocated to FCIC by the University.

Service:

FCIC is really a service organization. Requests for assistance and information from FCIC average about ten requests per day. Requests take the form of letters, phone calls, or visitations. The Center has responded to a letter from Hong Kong, and has hosted visiters from as far away as Manitoba, Canada.

In order to respond to the many requests, information sheets are develped for the more obvious types of questions. Many people request general information such as what journals are most appropriate for teachers or what printers are available for the Apple computer. In some cases requests are very specific and would require a great deal of research time. A key-word indexing system is being established to help find specific information in microcomputer journals.

Major Projects:

The Department of Education is supporting the FCIC for five major projects:

1. Develop a computerized index of quality software on the IBM 3033/N mainframe computer.

The Center is indexing software that is developed by major publishers, or is being used by schools or is of known quality. An item is indexed by publisher, subject matter, grade level, and type of computer. Additional information is also indexed that will indicate if the software package is available for preview or if reviews exist and who might be using the software.

The steps involved in indexing software involve the following:

a. Collect catalogs and existing index materials.
b. Code the software selected on index forms.
c. Keypunch to cards.
d. Submit cards to mainframe computer.

The Center is developing the capability of entering software information directly on a microcomputer and dumping the information to the mainframe. Presently Fortran and SAS are used to access the data and provide the Center Staff with specific information.

2. Aid the DOE in the construction and analysis of a survey of instructional computing in Florida.

A survey instrument was developed by the Center to determine the type of hardware and software used in the schools in Florida. The DOE futher revised and tested the form. Information from the survey should be useful to educators in planning and development of instructional programs involving computers.

3. Collect and write reviews of educational software.

The Center is collecting and indexing reviews of educational software. FCIC staff members and students involved in graduate programs in "Computers in Education" have reviewed a number of educational programs. FCIC has developed a review form to aid reviewers. Experience has shown that the review process is very time consumming. FCIC has also evaluated software using the procedure established by MICROSIFT.

4. Established and maintain microcomputer laboratories.

At the present time FCIC supports two different microcomputer laboratories, a teaching laboratory and a demonstration laboratory. The teaching laboratory has sixteen (16) Radio Shack Model III microcomputers in a network supplemented by nine (9) Model III's with single disk drives. The teaching laboratory serves the following groups.

a. Computer literacy classes.
b. Undergraduate classes in BASIC.
c. Graduate classes involved in degree programs.
d. Workshops for Teachers.
e. Noncredit continuing education classes.
f. Saturday gifted classes for students.
g. Summer Camp for junior high students.

The demonstration laboratory will consist of a variety of different types of microcomputers along with additional equipment such as video disc players, printers, voice synthesizers and optical scanner. The demonstration laboratory will be located on the top floor of the main library next to the computer science department and will serve a variety of needs.

5. Establish a computer software library and a collection of books and journals.

The software and book library will be located ajacent to the demonstration laboratory so users can try a variety of programs on different computers and locate the answers to specific questions. It is expected that most graduate research and development will be carried out in the demonstration laboratory. Several innovative hardware and software displays will be established in the laboratory.

Supplementary Projects:

FCIC is involved in a variety of services and research projects related to instructional computing and several projects are planned for the future. In the spring of each year FCIC sponsors a computer programming contest for senior high school students. Teams of four students each from several high schools around the state compete in writing programs on microcomputers. The contest is held at the Tampa campus of USF.

The College has been involved with classes in BASIC programming for junior high students and for gifted students. This summer FCIC will design and conduct two-week computer camps for junior high students. The objectives of the camp will involve a variety of unique activities related to computers.

FCIC conducts special workshops for teachers. This spring, business education teachers from the local school system will be involved in a specially designed workshop consisting of learning about BASIC and software packages appropriate for business education classes.

Staff members of FCIC are involved in research such as developing software to determine learning difficulties in mathematics, utilization of optical scanners as an input device, application of video disc technology, networking of microcomputers, development of message boards, use of touch tone phone input to microcomputers, and development of a simplified LOGO.

Future Projects:

Depending on the level of funding, additional projects of the Center will include:

a. Coordinate fifty (50) workshops for teachers by teachers to provide inservice training in computer topics at off-site locations.
b. Establish and maintain a microcomputer-based school management system in a school as a model for other schools interested in automating their record-keeping systems.
c. Conduct a summer software development project in which a selected team of teachers work with university consultants to develop quality educational software for a specific subject area.
d. Establish a programming contest for teachers to promote the development and sharing of good educational software.
e. Conduct a conference for Active Computer Educators in the State. (ACE Conference)

Center Organization:

The Center consists of one director, a hardware expert, a software specialist, an information specialist, and secretarial support. At the present time all of the staff are assigned part-time to the Center. The staff meets as a group once a week to discuss goals, issues and individual problems.

Technology is used to help manage the Center. For example, word processing, data management programs, an automatic telephone answering system, graphic printers, and a microcomputer message board are used. The Center is also active on the Source Telecomputing Service.

An advisory board on Instructional Technology has been established within the College of Education to provide input to FCIC, instructional computing programs and projects. Each month the Center performs a self-evaluation. The focus is on service. Points are awarded depending on whether the service performed was a letter response, talk, software review, article, workshop or conference. FCIC staff members have visited MECC, MICROSIFT, and Tandy Corp. to determine how other centers manage their activities.

Conclusion:

If FCIC is to become a viable information center for promoting the use of instructional computing in the schools, then a network of knowledgeable people around the state must be established. In order to keep up with the fast changing technology, the network must use some form of telecomputing. Information must be shared, organized and disseminated quickly and accurately.

The Florida Center for Instructional Computing and the College of Education at the University of South Florida both have a strong background in and commitment to instructional computing. FCIC has the opportunity to become a catalyst, as well as a resource, for the effective and creative use of computers in Florida's schools.

SELECTING EDUCATIONAL APPLICATIONS FOR MICROCOMPUTERS

Linda Perry Hausmann
Manager of Instructional Systems
EduSystems, Inc.

Computers are now a part of education. During the past three to four years, the advent of the microcomputer has put instructional computing power within the reach of most elementary and secondary schools. It is estimated that by 1985, between 80 and 90 percent of all school districts in the United States will be using some instructional computing. The rate of growth in the number of instructional computing devices in schools is expected to be at over ten percent per year.

The growth in hardware sales has opened up a market for good instructional software. Would-be authors are interested in producing software to meet the needs of education. Many small educational software companies have been started, and major educational publishers are anxious to have computer related poducts to offer their customers.

The level of enthusiasm and interest shown by the companies is very gratifying for proponents of computer use in education. However, overly zealous computer advocates can do more harm than good for the cause of effective use of computers in education. A common notion seems to be, "If material is good, it will be better on the computer, and if it is not good, putting it on the computer will make it better". Computers are no more a panacea for educational ills than were teaching machines and video tape.

While it's true that any device enjoys the benefits of being an attractive novelty when it's first introduced, those benefits are short-lived and eventually the product must stand on its own merit.

In authoring a package that is either completely computer-based, or that has a computer component, a crucial step is deciding which topics or portions of topics are most appropriately done on the computer.

In the development of many products a team of people are involved. The author selects the topics to be presented and gives general flavor and direction to the product. The instructional designer takes the basic ideas and tries to implement them as effectively as possible on the computer. The programmer follows the instructions of the designer. Any bad decisions made by the author can have a very adverse effect on the product. The author's mistakes "trickle down" through the product. Even the best design and programming cannot make up for a poor selection of topic.

Media Selection

A variety of media can be used in a given product. Authors should consider the strengths and weaknesses of several media when planning their products.

For centuries, books have done a wonderful job of presenting ideas, information, diagrams, charts, graphs, and pictures.

The learner has good control over the learning process when using a book. When bored, the learner can page ahead or read more rapidly. Similarly, learners who need a review can go back to an earlier section. Books have another important asset: they are wonderfully portable - you can carry a book almost anywhere. Books, however are not very good at providing feedback for the learner. (Programmed texts are a possible exception.) Also, most books are not designed to meet the needs of individual learners.

Films, video tapes, tape cassettes, and other similar media are very effective in the presentation of ideas and information. A good film is engrossing and its visual images are often remembered for years. But like books, films do not provide feedback, and they are not very effective for individualizing instruction.

The computer's greatest assets as a medium for instruction are its capability to provide immediate feedback to the learner, and its potential for learner control and individualization. The computer screen is not a particularly good medium for presenting written material. Given a choice, most people would prefer reading a book to reading a book's worth of information on a screen. Even a good monitor can be hard on the eyes it the user has to read an extensive amount of information from it.

Graphics are a wonderful new addition to computer technology, but some authors, in their excitement over this capability, use them ineffectively.

A graph or picture displayed on a screen is usually of poorer quality than one printed in a book. Graphs, charts and pictures showing static information are usually better displayed in a book.
The best graphs, charts, and pictures displayed on a computer screen are those that somehow use information entered by the learner.

Before developing a computer-oriented product, authors should ask themselves a key question:

> What can the computer do to enhance this product that could not be done through another medium?

If there is no good answer for this question, the product probably is not a good one to put on the computer, or at least it should not be implemented on the computer until a good answer to the question is found.

What are some guidelines for selecting products that could be implemented nicely on the computer? An answer to this lies in examining quality computer software, and determining how it benefits the learner. Below is a list of some of the common modes of using computers in instruction and a description of the characteristics and benefits of each:

Modes

tutorial
drill and practice
simulation
games
information retrieval
problem solving

In tutorials, the computer presents some information and then asks questions about what was presented. Good tutorials ask the learner questions frequently. As a rule, students should not see more that three or four screens of text before being asked a meaningful question. If a topic for a tutorial requires extensive reading before a good question can be asked, maybe the bulk of the text should be in a book that accompanies the package. Good tutorials have appropriate feedback for the various answers that students might give. They also make effective use of branching to move students to material that is easier, more difficult, or that better meets the students' needs.

In drill and practice, it is assumed that the student has already seen some presentation on the material and as the title suggests, just needs some practice on applying what has been learned. A well designed drill makes effective use of the computer's capabilies. Students can be given more work on skills they find particularly difficult or can skip drills they have mastered. The nature of drill and practice makes it easy to design good feedback.

Simulations let the student explore a concept through role playing. Students need to do some fairly high level thinking to be successful in using a simulation. They have to be able to look at several factors to arrive at a decision that will be most beneficial for the character they're playing.

A key thing to remember in designing or using simulations is that the computer simply uses the student responses as data for a pre-determined model. The author and designer do not have to think of the variety of student responses and have some appropriate feedback for each. The students receive feedback when they see the consequences of their actions. Being able to evaluate the consequences requires some higher level thinking skills, but the evaluation is done by the student, not the author.

Instructional games can be used in more than one way. They can be used to provide a motivating format for drill, or to teach strategy skills. If students need much drill on a particular concept, presenting the drill in a format where they can have some fun and entertainment is a good idea.

Some games are designed to teach strategy. Authors of this type of game should be careful not to replicate a perfectly good board game on the computer unless implementing it on the computer improves the game.

Information retrieval and problem solving are both applicatons where the student uses the computer as a tool to achieve his/her objectives. In many cases, this involves having the students write programs or use the computer to explore an existing database. In both cases, there is a high degree of learner control, and feedback - two of the computer's strongest assets.

Potential problems

The following are some "tip-off" comments that students could make when running a product that just doesn't "fit" on the computer:

"I want to go back about seven screens to review that fact."
- If the information were in a book that accompanied the courseware, the student could easily go back and review.

"Why do I have to do the problem this way? I think my way is just as good"
- Perhaps the author is attempting to do too much. Teaching students high level thinking skills is a very challenging task.

"I had better get out my pencil and paper to take notes on the computer's presentation."
- If the amount of text in a computer application is large enough to require note-taking, the author might want to consider putting the text in a book or manual.

"I had better get pencil and paper out to solve these problems."
- Either the problems were not divided into small enough steps for the students to be able to solve them on the screen, or perhaps the problems are not very appropriate for solution on the computer. Even in this age of technology, there are still some activities that are best done with pencil and paper!

In conclusion, authors should be very aware of how they plan to use the power of the computer. The computer can enhance many learning experiences, and be an integral part of a packages, but for some needs, it is less effective than a book, film, or best yet, a teacher!

MICROCOMPUTERIZED KAUFMAN WISC-R HYPOTHESIS GENERATION

The Wechsler Intelligence Scale for Children - Revised (WISC-R) is a commonly used instrument in the assessment of children's intelligence. Literally volumes of WISC-R research have been reported in many topic areas - cultural factors, psychoneurology, theories of intelligence, learning disabilities, gifted, and others. Unfortunately, because of the research to publication time lag, the various competencies of WISC-R practitioners, and the time required to review and integrate the relevant research, WISC-R users are often unaware of significant developments in WISC-R usage.

Several articles have recently addressed these problems, most notably perhaps in the series of "special" issues of the Journal of Learning Disabilities. Aside from the diversity of empirical information presented in these recent articles, they seemed to share one common theme - that the theoretical approach to WISC-R analysis presented by Dr. Alan Kaufman in his work, Intelligent Testing with the WISC-R, is the cornerstone of comprehensive and competant WISC-R usage. In turn, Kaufman's main tenet is that the primary use of the WISC-R should be the generation of tenable hypotheses as to a child's WISC-R performance.

As any practitioner knows, however, the time constraints and sheer number of cases with which the WISC-R user is faced often precludes the comprehensive analysis advocated by Kaufman. Even when "internalized", the clerical work involved in extensive WISC-R analysis can be of sufficient magnitude to discourage the practitioner from fully utilizing Kaufman's approach. As a result, then, the WISC-R practitioners who have the least time to review relevant research and commit to memory Kaufman's approach (that is, those practitioners faced with the most number of cases to be processed) are the very people who might most benefit from the routine employment of the Kaufman method. Obviously, there appears to be a gap between what is "ideal" and what is "practically feasible" in the area of WISC-R analysis.

One real solution to this problem lies perhaps in the application of microcomputer technology to Kaufman's method. As the cost of mocrocomputers decrease, their accessibility to and potential for usage by psychologists increases. Thus, the use of microcomputers to routinely perform Kaufman analysis (or rather, Kaufman hypotheses generation) appears to be a cost efficient, practical way to promote competent and comprehensive WISC-R analysis.

It is noted that such a computerized approach to WISC-R analysis would not necessarily have to generate a WISC-R report, as have some other computerized WISC-R approaches. (Indeed, this author is emphatically against the use of computers in this manner)! Rather, the microcomputer could perform the necessary calculations, "screen" the data for relevency to established empirical data on WISC-R performance, and present a list of case-relevant, comprehensively reviewed and empirically researched hypotheses that may be valuable avenues for further investigation. Whereas previous computerized attempts at WISC-R analysis have constrained the practitioner's intuition and expertise, the proposed approach could free the practitioner from the mundane clerical tasks, perform what are typically rather laborious literature reviews and thereby free the WISC-R user to more informedly employ his/her professional acumen. in addition, once Kaufman's theoretical approach were made computer-readable, it would be a relatively simple matter to enter significant WISC-R research findings into the computer memory structure and in so doing create a dynamic WISC-R data base which would be both comprehensive and current while allowing for routine reference by the WISC-R practitioner.

- Rich Ingram

MICROCOMPUTERS: ADMINISTRATIVE TIMESAVERS

Gary A. Kehrer and Nick A. Schepis
The School Board of Broward County
1300 West Las Olas Blvd.
Fort Lauderdale, Florida 33312

ABSTRACT

Based upon a needs assessment given to all school based administrators of Broward County, Florida, the following information was ascertained:

1) The number one ranked item was "COMPUTER LITERACY".

2) Schools are using microcomputers in their instructional programs and most thought they could be used to lessen the burden of administrative paperwork.

Given these parameters it was our task to develop what we thought to be a "COMPUTER LITERACY" program for school based administrators. Rather than conduct a course of programming in BASIC, we determined that given the tasks an administrator performs, a Data Base and a Word Processing package would better prepare them for using the microcomputer to solve administrative problems at their schools.

BACKGROUND

Broward County, Florida, the nation's tenth largest fully accredited school district, is unique in its approach to school-based management. The district is comprised of twenty-two (22) high schools, twenty-nine (29) middle schools ninety-seven (97) elementary schools, four (4) vocational centers and eight (8) exceptional educaton centers serving an approximate population of one hundred twenty seven thousand (127K) students. Inherent to school-based management is the operational and financial responsibilities downloaded from district to school site. This decentralization enables the school district principal to institute programs that serve their students. Fully aware of the impact of recent technological advantages in microcomputing, our principals readily financed a viable computer education program. This structureis now a part of the existing curricula in over 40% of the school district including 100% of high school and vocational centers: 50% of middle school and exceptional centers: and 20% of the elementary schools. Currently, there are over 450 microcomputers on site.

EVOLUTION OF INSTRUCTIONAL PROGRAM

First Generation (1965-71)

EDP-18 Computers at seven high schools. Advanced mathematics course for selected twelfth year students. Serviced 175-250 students per year. Each system cost $16,500.

Second Generation (1971-79)

Time-share system at six - ten high schools. Regular mathematics course for students completing first year of algebra. Serviced 250-400 students. Cost of each system varied from $6,000 to $18,000 depending on configuration.

Third Generation (1979-?)

Microcomputer systems (APPLE II) at 40% of schools in district. A sequence of courses of regular and advanced computer mathematics for students completing Algebra I. Involving 1250 - 1400 students in the program yearly. The cost of each system varies from $1,500 to $6,000 depending upon peripheral devices.

ADMINISTRATIVE INVOLVEMENT

Due to the tremendous impact and visible success of the microcomputer in the instructional program, school administrators became fully aware of the potential worth of the microcomputer as an administrative tool. This awareness, coupled with the unreal expectations of the computing and storage capabilities of the microcomputer created a need for a serious evaluation of microcomputing relevant to administrative applications.

INITIAL PROBLEMS AND RESOLUTIONS

1. An undue burden was placed on the computer math instructor to write applications code for the administration.

   This problem resolved itself as the computer math instructor delegated this chore to the advanced student as an independent study task.

2. Demand for elaborate file management imposed limitations as to the size of a manageable file. For example, an attendance program written for a school of an enrollment of seven hundred (700) students on single session would become incompatible with a school of thirty-five hundred (3500) students on double session.

Alleviation of this difficulty recently occurred as several reputable data-base systems appeared on the market. The following systems have been used successfully by schools of varied enrollments:

A. CCA DATA MANAGEMENT SYSTEM by Personal Software, Inc.
B. DB MASTER by Stoneware Microcomputer Products.
C. INFORMATION MASTER by High Technology, Inc.

3. Repetitive computations having to be made by the administrator to generate a variety of reports was quite time consuming and affected inefficiency and inaccuracy. For example, a county level administrator in charge of employee relations had to spend hours computing various salary schedules; a school-based administrator having to constantly recalculate the school budget.

   It was possible to correct such inaccuracies and promote efficiency through the use of a microcomputer linked with a variable electronic calculator software package. We found these traits in VisiCalc by Personal Software, Inc. Not only was it a timesaver, but an added dimension was apparent when the administrator was given the ability to play "WHAT IF..." simulation with salary schedules; a school-based administrator having to constantly recalculate the school budget.

4. School administration realized that dedicated word processing systems would increase the productivity of their clerical staff. Costs of these systems were prohibitive to an individual school's budget.

   Since microcomputers existed at these sites, an inexpensive and reliable software package was needed. While WordStar by Micro Pro International, Inc. offered the closest to what was needed, the expense of the software along with additional hardware needed for proper interfacing was unjustifiable to most school budgets. Several inexpensive word processing systems were tested; although many were suitable, Magic Window by Artsci, Inc. was selected.

PROBLEM SYNOPSIS

Software packages exist for all facets of administrative tasks. The following list summarizes those tasks as implemented in Broward County:

Data-base packages are used for:

1. Activity calendars
2. Athletic schedules
3. Athletic statistics
4. Attendance lists
5. Course offerings
6. Equipment inventory
7. Master schedules
8. Media center statistics
9. Referral statistics
10. Requisition information
11. School calendar
12. School facilities
13. School keys
14. School parking
15. Teacher schedules
16. Textbook inventory

VisiCalc is used for:

1. Enrollment projections
2. F.T.E. preparation
3. Salary schedule simulations
4. School budget

Word processing is used for:

1. Daily bulletins
2. Faculty correspondence
3. Newsletters
4. Handbooks and manuals
5. School reports
6. School calendars
7. School surveys

This list is by no means complete since specific applications are only limited by need and creativity of the administrator. This process gives new meaning to the concept of "Computer Literacy".

COMPUTER LITERACY REDEFINED

As one scans the current literature in search of a definition for Computer Literacy it appear that most institutions use some type of BASIC programming as the nucleus. We submit that to spend the time to instruct administrators in the intricacies of programming is neither productive nor cost-effective. Our definition of Computer Literacy takes a new meaning. In essence, to be able to operate a system, utilize existing software, and maintain both in good working condition is Broward County's conceptual definition of Computer Literacy.

Inservice instruction for school-based and county level administrators is currently conducted through the Broward County Management Academy. Through a lab setting, "hands-on" experience and a working knowledge of the microcomputer system and supportive software is made available to each participant. School principals, administrative assistants, county level administrators and staff become fully aware as to the power of the microcomputer as an administrative timesaver.

CONCLUSIONS

The microcomputer is a viable and reliable administrative tool. Broward County has found that the APPLE II PLUS 48K double disk drive microcomputer system to be exceedingly cost effective. With well over four hundred fifty (450) systems in use, both instructionally and administratively, the maintenance costs are averaging less than fifteen (15) dollars per system annually.

These systems include external peripherals such as monitors, printers and modems. Software is dependable and user-oriented. As the number of users increase, so will the number of applications. Therefore, system and software enhancements are constantly being evaluated. It is still the enthusiastic and creative nature of the individual administrator who develops the time-saving potential of the microcomputer.

FLORIDA'S INFORMATION RESOURCE NETWORK

-Alan T. Olkes

As the eighties are upon us, one fact stands out clearly: past practices and procedures in data processing are inadequate to meet the challenges of the future -- in short, to manage the most critical resource of the eighties: INFORMATION! While these challenges are not totally new, the magnitude of them most definitely is.

Several factors cause this to be true. Computer hardware continues to cost less and do more, which widens the scope of potential applications. This, however, is limited by a shortage of qualified personnel to develop them. These problems are then compounded by the great increase in the number of users and their wide dispersion of data processing knowledge.

The concept of information resource management (IRM) is the recognition that information is the lifeblood of today's organizations and the resolve to manage this resource just as other critical resources are managed.

Before something can be managed, however, it must be defined. Therefore, the first step must be "Data Definition." The Florida Department of Education has made an initial start, in the establishment of data element dictionaries which are continually being updated. These serve as the basis or hub of an IRM.

The second step involves the dissemination of information from its source to the agent needing it. This must be handled by some type of automated network, tying together schools, districts, the State and, at some future date, the Federal government. The development of this information network, called Florida Information Resource Network (FIRN), required critical future thinking to insure that it not only meets today's needs but that it is flexible enough to evolve with the ever-changing data processing environment.

The dissemination of the information resource through an integrated network directly increases the productivity of the recipients by enhancing their decision-making process. It gives them the tools to do their jobs. No longer will non-data-processing professionals be kept in the dark as to what information is available, how current it is, who is responsible for it, how may it be accessed and so forth.

The 1981 Florida Legislature provided $175,000 to the School District Council on Comprehensive Management Information Systems, a group representative of the Management Information Systems section of every school district in the State.

The Council, working with the Department of Education, in addition to its normal duties, developed a plan for the implementation of the Florida Information Resource Network (FIRN). The plan, submitted to the Legislature January 15, 1982, provides for an organized process of identifying information needs, collecting and processing data, and providing information for decision-making. Information will be provided in a timely fashion and in a form which aids understanding and stimulates action.

The plan leads, through a phased implementation, to an automated delivery system of information, flowing from individual schools through the District to the State and back.

°Phase 1 provides semi-automated capability at the district and/or regional level, using magnetic tape transfer or data communications transmission to the State Department of Education. This phase is complete as of this date.

°Phase 2 provides automated capability at the district and/or regional level using magnetic tape or diskette transfer with increased data communications transmission. During 1982, terminals, microcomputers or other such equipment will be installed in the districts without automated reporting capability. A second activity will focus on further defining the data elements, logical design and physical implementation of the required "pool" of student, program, staff, finance, facility and community information.

An application software development team will be established to do a user needs analysis and develop a logical design of the information base hierarchy, level by level. In addition, procedures will be developed to promote a more compatible information processing hardware base, throughout the State.

The final activity of the phase will be the initiation of a "Network Pilot Project," a communications link of three areas in the State. This pilot will test the abilities of vendors to facilitate the transmission of data to and from various brands of computers. This phase should be complete by August of 1983 at a cost of $1,488,000.

°Phase 3 provides automated capability at the district and/or regional level with batch bi-directional data communications transmission of structured information. All 67 districts will be linked to the State and each other, either directly or via a regional host, to enable electronic transmission of information in a bi-directional manner. This phase will begin in 1983 and is scheduled for completion in August, 1984. The estimated cost is $8,129,000.

°Phase 4 provides interactive ad hoc information retrieval and reporting capability for district, state and legislative levels of educational management. Scheduled to begin in 1984, this phase should be complete in early 1986, at a cost of $14,578,000.

°Phase 5 extends the previous phase down to the school level, with terminals in every one of the 2,345 public schools. To be implemented in 1986, this phase will cost an estimated $21,155,000.

The plan presents the design for such a network and a discussion of the seven individual resources (Network Hardware, Network Software, Network Technical Staff, Information Base/Data Review Committee, Information Processing Hardware, Information/Application Software and Information/Application Technical Staff) which comprise the final product and are critical to its success. These resources should be understood individually yet viewed as a whole for the plan to succeed. In presenting the design, each resource is introduced as an analogy to provide an understanding of the function served by the resource without using technical terminology.

With the purpose of an information network being to provide the ability to deliver the right information at the right time to the right place as efficiently as possible, it can be likened to the development and operation of a railroad.

Prior to laying track, railroad planners must assess the need for delivery to various locations, anticipated volumes and frequencies, etc. In the Florida Information Resource Network (FIRN) these "destinations" might be districts, schools, regional centers, or the Department of Education.

Planners must also know the various characteristics of the "information freight" to be delivered - how much, how often, and in which direction. This knowledge - how much information freight originates at each depot (node), how it must be packaged, to where it is delivered and how often - is the design of the Information Base itself.

Equipped with this information, just as the railroad planners can design the network of tracks which will connect each of the depots, the network planners can connect each of the nodes in the FIRN.

The planners recognized the need to have connector lines supporting the daily flow of information freight between school depots and their respective district offices as well as long haul lines which could carry larger volumes of freight over longer distances. They must install switches also, allowing trains to be routed to different destinations when the anticipated volume of freight could not justify the cost of individual tracks. These tracks and switches are representative of the Network Hardware required to establish routes for the information to travel between points.

With the tracks (network lines) laid, the network is in place. The tracks do not dictate what brand of train runs on the tracks. However, each train must be of the same gauge as the tracks, be able to clear bridges, and so on. This is also the case with FIRN. Information may originate on any kind of Information Processing Hardware, as long as that computer can "fit the tracks."

Now the network is ready to move to its next phase--operation. It is immediately apparent that two trains cannot head towards one another on the same track without colliding. Also, switches must be thrown to direct them to the correct destination. Therefore, it is necessary to have a system of signals to control routing and scheduling. With the FIRN, this is the function of the Network Software. This resource of the FIRN controls the movement of the information.

The network is now equipped to operate but still unable to deliver its information freight. Somehow the freight must be loaded on the trains. This function is met by Information/Application Software which has been designed and programmed by Information/Application Technical Staff. These are technicians who are knowledgeable of system design, programming and the user procedures performed by educational administrators. They can be selected from within the educational system and/or private firms to form the most proficient team possible to address the task at hand. The Information/Application Software will maintain a "dynamic" Information Base of Student/Program, Staff, Finance, Facility and Community data. An information request can now be initiated in Tallahassee, sent down the network communication lines and received at a network node. Information/Application Software will then analyze the request, gather the information needed to effectively respond, package the information, affix a destination address and load it on the next train (packet) leaving for Tallahassee.

Finally, the network appears to be operational yet no information is being transmitted. The last resource is missing. To function, the network must have Network Technical Staff just as the railroad must have engineers and switchment. These persons will actually operate and maintain the network and see that the "information freight" is delivered to the right destinations at the right time.

The benefits to be realized with the completion of the full plan include -

Teacher Data Burden Reduction - promoting the reduction of teacher paperwork through the use of computer hardware and software at district and school levels for activities such as student record keeping, grade reporting and attendance reporting.

Automated Reporting - enabling the transmission of information from the State to the school districts/schools and vice versa.

Record Keeping - storing student records one time in machine readable format, enabling the transfer of information from one institution to another, i.e., school to school, district to district, district to community college, district to university or community college to university.

Instructional Computing - establishing the foundation for the integration of administrative and instructional computing to support activities such as a Computer Managed Instruction (CMI). Also, making the network available for file to file transfers and down-line loading in support of Computer Assisted Instruction (CAI) and Computer Science courses.

Accuracy and Timeliness - improving accuracy and timeliness of information at all levels within the state-wide educational system.

Automated Procedures - increasing automated user functional procedures at district and school levels, thereby providing greater efficiency in handling student/program, staff, finance, facility and community information.

Resource Sharing - providing a vehicle for sharing data processing resources (people, hardware and software) among school districts, community colleges and universities.

Equitable Access - enabling more equitable access to functional computing (computer hardware, systems software and applications software), regardless of the size of the school district.

Cost Reduction - furnishing modern telecommunication technology as a hedge against the ever-increasing cost of conventional data communication methods.

Innovative System - making a working telecommunication prototype available as a model for use by other governmental agencies.

Non-Data Processing Co-op Model - establishing a model for educational entities to share non-data processing resources, such as audio visual, in-service training and group purchases.

Electronic Mail - providing the potential for greatly improving the speed at which correspondence travels from one educational entity to another and reducing the cost associated with this document transfer.

For a copy of the complete plan, write to:

Edward R. Allen, Jr., Administrator
M.I.S., Fla. State Dept. of Education
275 Knott Building
Tallahassee, Florida 32301

and request a copy of:

"Florida Information Resource Network"

It Does Not Have to Be a Black Box:

The Creative Use of a Microcomputer As A Computational Aid In The Educational Environment

By
Eric I. Pas
and
Edward Anapol

Department of Civil and
Environmental Engineering
Duke University

ABSTRACT

Speed and accuracy make the digital computer highly desirable for undertaking complex and repetitive calculations. Recently, there has been increased use made of the computer in the educational environment, both as a problem-solving aid as well as in the drill and practice, tutorial, and simulation modes. However, as a computational aid in the educational environment the computer can be disadvantageous by reducing the students' need and motivation for understanding basic principles and problem-solving techniques. This paper describes the development and evaluation of computer software designed specifically for use as a computational aid to enhance the educational process. We show that it is both possible and desirable to use the computer in this manner, and we develop general guidelines for designing such software.

INTRODUCTION

Digital computers are used widely as computational aids. They are suited particularly for complex and repetitive calculations because of their speed and accuracy. The computer is increasingly being used as a learning aid, primarily because of the recent availability of low-cost microcomputers often having color graphics capabilities and a variety of software. The potential use of the computer in the drill and practice, tutorial, and simulation modes is generally recognized. However, the use of the computer as a computational aid in the educational context is more problematic, because the computer can eliminate the students' need for a thorough understanding of basic principles and problem-solving techniques.

This paper describes the development of a set of computer programs specifically designed to be used as a computational aid while enhancing the learning process. First, we briefly describe the particular application area and the characteristics which make the computer potentially useful. Second, we discuss the objectives of the computer assisted learning (CAL) package and the guidelines used in its development. Third, we evaluate the performance of the CAL modules. The paper ends with a summary of our conclusions.

CAL PACKAGE DEVELOPMENT AND IMPLEMENTATION

The planning of urban transportation systems has for the past 25 years been undertaken using mathematical models representing the demand for and supply of urban transportation services. These models generally require numerous inputs, have numerous outputs, and often involve repetitive and/or iterative calculations. Thus, a computer is essential in applying these models.

The models used in transportation systems analysis are somewhat abstract and civil engineering undergraduates often experience difficulty in understanding and applying them. In order to familiarize students with the models and the transportation planning process it is important to allow them to apply the models. Even if the problem is reduced in scope, many tedious and repetitive calculations must still be performed. While it is desirable for the students to perform each set of calculations once to gain an understanding of the techniques, repetition of the calculations is merely time-consuming and detracts from the overall learning process. In particular, the effort expended in numerous calculations can detract from the students' understanding of the models themselves, as well as preventing an appreciation of the important system interactions that can be highlighted through application of the model system. Furthermore, any time not spent on tedious, uninformative calculations can be spent better in examining the impacts of alternative transportation systems for the study area.

At Duke, we employ a case study of a small hypothetical area in order to familiarize our students with the travel forecasting models as well as to give them an appreciation of the important system interactions. The case study project is undertaken by groups consisting of three students. The overall goal of this research was to develop a CAL package that would aid the students in undertaking repetitive calculations, without eliminating the students' need for understanding the basic principles and techniques being employed. That is, we wanted to develop a computer package that could not be used as a 'black box.' We hypothesized that by eliminating the need for repetitive and time-consuming calculations, the CAL package would in fact enhance the learning process, if it could not be used as a black box.

Several guidelines were developed and employed in the design of the CAL package. The most important of these, and the one which best distinguishes this package from those used by professional transportation planners, is that user competence and understanding are not assumed. In fact, we assume that the user is inexperienced and is relatively unfamiliar with the principles and techniques. Thus, the user is required to demonstrate a thorough understanding of the models before being able to employ a module as a computational aid. The package is designed for use with a limited range of problem types, in contrast with the flexibility required of industrial software. The package is designed to be easy to use and is interactive, in order for the user to obtain rapid feedback. The modules are programmed in BASIC for the IBM 5100 microcomputer. The package is designed for use in a computer laboratory by small groups, and sessions at the computer are expected to be between 25 and 35 minutes in duration. Instructions should be clear and unambiguous, and diagnosis of a user's difficulties with a given model is undertaken where possible. Help, in the form of a review of a particular model, is provided on-line for those users experiencing difficulties. These guidelines were most useful in designing the CAL package discussed here.

The tasks to be performed in the transportation planning exercise are divided into several modules, each module is designed to constitute a single session at the computer. An introduction module is included in the package to provide supplemental instructions on the use of the CAL package and to guide the student in the performance of the subsequent planning tasks. Each module includes a number of questions to determine the user's understanding of the planning process and model system. In particular, the input and output variables for each module are examined. An incorrect response automatically generates the correct answer as well as additional information concerning the modeling technique being employed.

After completing the question segment of a given module, the student is asked to input the data required for the calculations and to provide first series of results. These results are evaluated relative to the input data provided by the student. If the calculated results are incorrect, the student is so informed and is provided another opportunity to perform the calculations manually. Correct responses enable the student to proceed with the CAL package. In subsequent sessions, the student who has successfully completed a given set of calculations manually is able to use the computer to perform these calculations, by entering a secret password.

As an example of how this package overcomes the problem of students using a program as a black box, we refer to a specific set of calculations undertaken in the urban transportation planning case study for which the CAL package was designed. At one stage it is necessary for the students to iteratively adjust the rows and columns of a matrix until a closure criterion is reached. An industrial-type package would require the input matrix, and possibly information regarding the level of closure desired, and would provide the final results. That is, the package is 'result-oriented.' Our module for this stage in the forecasting process requires the user to undertake the first set of the iterative calculations manually. After this step is successfully completed, the module performs the subsequent calculations. However, after each iteration the user is required to decide whether to undertake a further iteration or not. In this way we ensure that the student understands the closure criterion, and also that he/she obtains the benefit of closely observing the results at each iteration. The latter is a benefit that often derives from manual calculations. The module is therefore able to provide many of the insights usually obtainable only through manual computations. That is, our CAL package is 'process-oriented,' while eliminating tedious and repetitive calculations.

EVALUATION

The performance of the CAL package was evaluated both formally and informally. The formal evaluation took the form of a detailed questionnaire distributed to the students immediately following the completion of the case study. The informal evaluation of the modules' performance consisted of observation of the students as they used the CAL modules, informal discussions with various students, examination of the types of questions asked of the instructor, and the scope and contents of the analyses undertaken in the case study project.

The questionnaire consisted of 50 questions and many of the reactions were obtained via the use of a five-point Likert scale ranging from 'strongly agree' to 'strongly disagree.' Statements were worded both positively and negatively. Based on the responses of the 17 (of 30) students in the class who completed the questionnaire, a number of observations can be made. Student reaction to the CAL package was generally very positive. More than 70 per cent of the respondents found the modules useful mainly for simplifying the computations. However, only 18 per cent of the respondents found the modules to be of no help in improving their understanding of the transportation planning methods and in formulating questions to be discussed with the professor. Furthermore, only 12 per cent felt they would have learned as much as they did from the case study project without the CAL package, while about 40 per cent of the respondents appear to have used some of the modules for personal review of the material. Only one respondent felt that the CAL package rendered the instructor unnecessary, and about a third of the respondents felt that the help provided should have been more detailed.

The informal evaluation of the CAL modules supports the above conclusions. In particular, we observed a great deal of interaction between group members when they were using the CAL modules. This is in sharp contrast to the usual pattern of group interaction when discussing the case study with the instructor. In the latter case, it is common for one individual to participate in the discussion with the instructor, while the others merely are present--perhaps for moral support. It is likely that the computer engages the students in an "us versus the machine" situation, and it is also possibly less threatening than the instructor, in that the machine will not be evaluating their performance for the purposes of assigning a course grade. Informal discussions with students during the course of the project not only indicate the value of the CAL modules, but also provided information regarding changes needed.

The success of the CAL modules is also indicated by two other observations. First, some groups undertook analyses this past semester that were not attempted the previous two times the case study was assigned without the availability of the CAL package. This was probably due to the reduced amount of time spent on the calculations, as well as the greater insights gained. Second, the course evaluations in the past have shown that the students generally find the project interesting and didactic. However, the past complaints about the large amount of time spent doing repetitive calculations were not present this time.

SUMMARY

This paper describes the development and evaluation of a CAL package to be used as a computational aid in the educational environment. Our experience shows that with very careful design one can develop a problem-solving aid that is not a 'black box' which would detract from the overall learning experience. In fact, we found that the CAL package described in this paper actually enhanced the students' understanding of the basic principles and techniques. In designing such software it is most important to recognize the differences between a computational aid intended for use by students and one to be used in industry. In particular, the latter is 'result-oriented,' while the former must be 'process-oriented' and should not assume user competence and understanding. Furthermore, because a problem-solving tool for the educational context need not be capable of solving a very wide variety of problems, it can be designed so that the user need not spend a large amount of time learning how to use it.

122

LEVELING COURSES FOR STUDENTS PURSUING A MASTERS DEGREE IN COMPUTER SCIENCE

Dr. Kathleen M. Swigger and Dr. Fred J. Zamecnik

This paper discusses a variety of issues concerning the proper design of a series of courses that can best be described as a "leveling" sequence for would-be masters of computer science students at North Texas State University. The issues discussed here are helpful to graduate, undergraduate and junior college programs because of the mutual desire of all three groups to find an effective and efficient program for introducing data processing concepts to a large number of students. The observations cited in this paper can be extremely helpful to any educator who is interested in developing similar programs of study.

STUDENT POPULATION

At the outset, it is necessary to recognize that the student population enrolled in most Masters Degree programs has drastical changed over the last five years with respect to background, motivation, expectations, and analytical skills. A vast majority of the masters students are returning to college in order to retool for major career changes. The lucrative employment opportunities in computer related jobs have attracted persons from varied disciplines. Some of these students, therefore, have not taken a mathematics course since high school or their freshman year of college. Some of the students wish to learn "just" programming, while others want to know about the use of the computer in a specific area. Some students have "played" with computers either at work or at home, while others have no experience with computing at all. These variations complicate the teaching process and make the design of a suitable series of "first" courses difficult; no single path is likely to meet the needs of the majority of students and compromise is inevitable.

Previously, masters students enrolled at North Texas who did not have the appropriate undergraduate background were required to make up deficiencies by enrolling in 21 hours of undergraduate core which included an Introductory course, PASCAL, PL/I, Assembler, Systems Analysis, Data Structures, and a Logic Laboratory. As the student population in computer science increased over the years, it became apparent that this undergraduate sequence of courses was not satisfying the needs of the either the graduate or undergraduate student population. Our undergraduates felt uncomfortable with the large number of graduate students in their course and, more importantly, the graduate students were faced with a 3 or 4 year degree program.

The Computer Science faculty at North Texas resolved this problem by designing a three-course sequence that integrates all the concepts presented in six undergraduate courses. The first courses teach the major programming languages of (IBM 370 Assembler and PASCAL and PL/I) along with machine structures, and systems analysis. Both courses attempt to focus on the design of algorithms within the context of a specific programming language. The third course combines concepts from Data Structures and advanced programming. The last two courses can be taken concurrently,since they are designed to complement each other and provide reinforcement where necessary.

CURRICULUM STRUCTURE

There are no computer science prerequisistes for the first course in the sequence. For an overall view of the program, a brief outline of each course follows:

Computer Science 503: PASCAL Programming and Machine Arithmetic Topic Outline Class Hours

1. Computer Applications. 3
History of the Computer. Computer uses in business, science, and real-time situations.

2. Algorithm Formulation. 4
The problem definition phase. An algorithmic language. Formulating and developing algorithms. Efficiency of algorithms. Recursion.

3. Number Systems. 5
Positional representation. Decimal, binary and other radix numbers. Conversion between radices. Signed-magnitude and complementary notations. Arithmetic algorithms in computer processors. Floating point representations.

4. Elementary Programming Concepts. 9
The concept of data types, names and variables. Arithmetic and boolean expressions. Assignment statements, input and output statements. Structured programming.

5. Computer Logic. 5
Switching algebra, logic gates, functional logic subunits.

6. Data Types and Structures. 4
User-defined and subrange data types. Structured data types, including arrays, records and sets. Pointer types.

7. Computer Structure. 14
Computer architecture, arithmetic units, memories, control units, input and output units and their interactions. Register transfer language.

8. Control Structures. 8
The compound statement. Three types of repetitive statements. Conditional statements. Declaration and invocation of functions and procedures. Recursion.

9. Program Debugging and Testing. 3
Syntax, run-time and logical errors. Program testing and validation. Documentation.

Text Books:

Schneider, G. M., Weingart, S.W., and Perlman, D.M. *An Introduction to Programming and Problem Solving with Pascal*. John Wiley and Sons, 1978.

Abd-alla, A.M. and Meltzer, A.C. *Principles of Digital Computer Design* Prentice-Hall, 1976.

Computer Science 504: PL/I and Assembly Language Programming.

Topic Outline Class Hours

1. Problem Solving. 8
Introduction to structured programming. Program debugging and validation. Good programming techniques. Documentation.

2. Elementary PL/I Programming. 9
Discussion of PL/I statement types: the assignment, I/O, conditional, and iterative statements. Character manipulation and file processing.

3. Data Types and Structures. 9
Specification of data types. Vectors, arrays, hierarchical data structures, tables, pictures. Character data.

4. Subroutines. 4
Program modularity. Intrinsic functions and libraries. User-defined functions and subroutines. Internal and external modules. Global and local variables.

The parameter linkage.

5. Introduction to Assembly Language Coding. 7
Relationship of ALC to higher level languages. Review of number system and machine arithmetic concepts.

6. Data Representation. 4
Numeric and coded data. Arrays, vectors and tables. Storage allocation for data.

7. Assembly Language Coding. 12
The syntax and semantics of assembly language statements, including move, compare, print, register-to-register and branch statements. Indexing.

8. Advanced ALC Concepts. 4
Declaration and invocation of subroutines and macros.

Text Books:

Hughes, J. K. PL/I Structured Programming John Wiley and Sons, 1979.

Stern, N., Sager, A., and Stern, R. A. 370/360 Assembler Language Programming John Wiley and Sons, 1979.

Computer Science 505: Computer Systems Analysis and Data Structures

Topic Outline Class Hours

1. System Configuration. 7
Comparison and analysis of hardware. I/O devices and storage devices capabilities. CPU comparisons.

2. Software. 3
Programming techniques such as top-down design, structured programming, and pseudocode. Comparison of languages. Operating systems. Computer files and data bases.

3. System Analysis. 4
Life cycle of information systems. System analysis and design stages. Preliminary surveys, feasibility studies and comparative analyses.

4. Advanced Concepts. 4
Multiprocessing, multiprogramming, virtual storage, data communications, and computer security.

5. System Operation. 3
Analysis and operation of a minicomputer system. Tape, disk and printer operations. Operator control functions.

6. Team Projects. 6
Research in a computer science topic. Oral and written report.

7. Static Data Structures. 6
Logical and physical aspects of data structures. Vectors, arrays and records. Storage representations and addressing mechanisms for static structures.

8. Semistatic Structures. 4
Problems associated with variable bounds arrays. Applications and representations of stacks, queues and deques.

9. Dynamic Structures. 11
Coverage of forward, circular and bidirectional linked lists, lists, multilists, general and binary trees and graphs. Insertion and deletion algorithms and tree traversal.

10. Searching and Sorting. 3
Search and sort algorithms with emphasis on the data structures. Comparison of sort, insert and retrieval algorithms.

11. Language Considerations. 3
Comparison of languages with respect to data structures.

Text Books:

Horowitz and Sahni, Fundamentals of Data Structures, Computer Science Press, 1976.

Bohl, M., Information Processing, SRA, 1980.

The decision to select particular programming languages and topics was the result of expedience and tradition rather than any truely scientific discussion. PASCAL was chosen as the language for the first course because of its modularity, 'readability and "high-level " concepts. Also, North Texas State University is fortunate to have a large Microcomputer Laboratory that can facilitate student programming projects in PASCAL. PL/I provides a high-level language capable of introducing more advanced language features and semantics along with runtime error checking.

Assembly language and machine structures are included because they acquaint the student with computer organization, providing insight into machine language, data representation, and computer architecture. It will also give the student a marketable skill in commercial programming. This course emphasizes machine problems to be done by the individual student.

Finally, the student is exposed to advanced data structures and advanced programming techniques. Using a high level language, the student learns to write programs designed to illustrate the implementation of selected data structures, the importance of the design phase of the software cycle and software engineering considerations.

EVALUATION

There were no formal mechanisms for evaluating students who completed the leveling sequence. However, some informal observations are as follows. Out of the 120 students who enrolled in the course during the first year, 60 students completed the sequence. More than one-third of the students each semester were registered for two of the courses at the same time. Also, 30 of the students enrolled in the leveling courses remain graduate students today. On completion of the leveling course sequence, many students are able to find immediate employment.

Because of the compression of time, many concepts that are normally introduced during class discussion are not available to these students. As a result, some of the students feel that they are at a slight disadvantage. However, the students who finish the leveling sequence courses compare favorably to students who enter graduate school through the normal admission process.

CONCLUSION

In retrospect, North Texas State University Computer Sciences department appears to be satisfied with its experience with the leveling sequence concept. The course content and emphasis differs only slightly from the undergraduate offerings. However, because the students are older and highly motivated, they are able to advance at the more rapid pace. Clearly, as much is learned and experienced by the students in the leveling sequence as in the regular undergraduate courses. But it is also clear that some students feel rushed, that there is not enough time for "playing around" on the computer. Also, not as much time is available for an array of different applications. The natural maturation process that occurs during normal coursework is not present for the students enrolled in the leveling courses.

The design of a series of courses as a leveling sequence is an important issue; because these courses offer the industry a mechanism to increase production of computer science majors. This paper has identified a variey of factors affecting the design of such a series of courses; any recommendations made, however, are meant to be not the final wordbut merely an attempt to define important issues.

THE DESIGN, DEVELOPMENT AND EVALUATION OF A LOW-COST COMPUTER MANAGED SPELLING SYSTEM

Rex Thomas and Brian Gustafson

Iowa State University

INTRODUCTION

The purpose of this project was to demonstrate that a low-cost, time-shared computing system could serve student needs in the classroom setting. The project involved the development, construction and evaluation of a micro-based computer managed spelling system. The system was designed to incorporate the most advanced technologies with sound instructional practices. It was evaluated in a surburban elementary school.

The subject of spelling was chosen for the project because of the nature of the subject and the recent technological breakthroughs in speech synthesis. Traditionally, spelling instruction has been either a very labor-intensive process of adopting word lists to individual learner needs or else a very inefficient process of assigning the same list of words to all students. The advent of advanced speech synthesis and economical mass data storage capabilities invited a technological solution to this persistent and important problem.

SYSTEM DESIGN

A system was developed consisting of a Texas Instruments 99/4 Personal Computer, a 99/101 micro-computer, a printer, and 8 Speak & Spell units. A student could type his/her name on the personal computer and be assigned to one of the 8 Speak & Spell terminals to spell the selected words. The personal computer maintained on disk storage a record of each student's progress as well as a list of 792 words. When a student signed-on, his/her record was retrieved and a list of words was selected based on a history of the student's performance. This list of words was transmitted to the micro-computer which retrieved from disk the digitized speech data for the words. This speech data was transmitted to the Speak & Spell terminal where the student heard and spelled each word. A record of the student's achievement was transmitted back to the home computer where the file was updated and the list of words, appropriately scored, was printed for student and teacher use. The system configuration is shown in Figure 1.

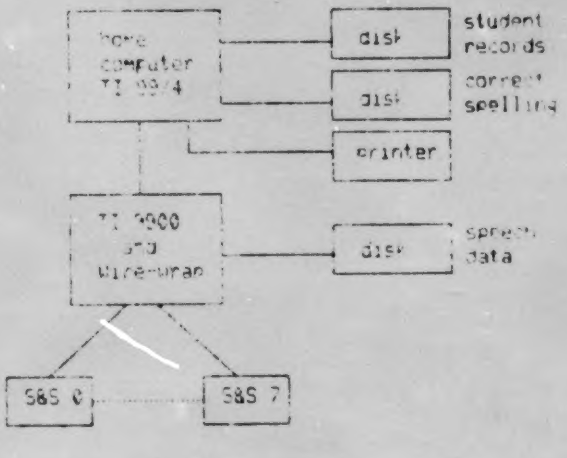


Figure 1. System Configuration

The algorithm for word selection was based on a state diagram. Words which a student had not seen were in state zero. They were moved to state 1 if they were spelled correctly and to state 2 otherwise. State 3 was reserved for words which had been spelled correctly two times and words were moved to state 5 if they were spelled correctly three times. State 5 words were assumed to be known by the student and were not presented again. States 4 and 6 were used as feedback loops for words which were spelled incorrectly.

RESEARCH DESIGN

An experiment was conducted over a 6 week period in the spring of 1981.

This project was funded through a Texas Instruments Contract.

The spelling machine was set up in one end of a large classroom adjacent to a paraprofessional station. One of the duties of the paraprofessional was to monitor the spelling machine and correct equipment problems. Assisting students with spelling questions other than those involving equipment was discouraged.

The students assigned to the experimental group were given two training sessions to familiarize them with the use of the equipment prior to the beginning of the experiment. During these sessions the students learned to sign-on the machine, find the assigned station, use the Speak & Spell units and inspect their progress report.

Students in the experimental group were scheduled to use the spelling machine each day. A student was assigned to 10 or 20 word sessions depending on the teacher's and experimentors estimate of his/her spelling ability and the amount of time available. As students became more proficient at using the equipment, more students were assigned to 20 words per daily sessions. After each session, the students were given a list of the words seen with an indication of their success on each word. During the experiment no tests or supplementary learning activities were conducted.

The students in the control group were told that the machine could not accommodate all of the students in the school and that their turn would follow. They were also told that the students had been randomly assigned to the groups. The control group students contracted with their teacher the number of lists of words they wanted to spell each week. The students were given the lists on Monday and practiced the words each day. On Friday the students were paired so that students could read the lists to each other for testing. If more than two words on a list were missed the list was repeated the next week, usually with the addition of a new list.

Following the primary portion of the experiment, all students were given a 40 word test consisting of a random sample of the words they had seen during the experimental period. The percent correct as well as the beginning word and ending word for each student were recorded.

The students in the control group were then assigned to the machine for the next 5 weeks while the students in the experimental group resumed their regular spelling program. At the end of this period the previous test was readministered as a retention test to all students.

Following the primary experiment when the control group was assigned to the machine, 44 students were randomly selected to determine the effect of the number of words per session on achievement. One-half of these students was assigned 30 words per session and the other half was assigned 20 words per session. After 5 weeks an achievement test consisting of 40 words was administered to both groups. As in the primary experiment the words were randomly selected for each student from the list of words the student had seen.

EXPERIMENTAL WORDS

There were two spelling programs in use at the school prior to the experiment. One was a textbook approach and the other was a contracted list approach. The contracted lists were divided into levels A through R with 9 lists in each level. The number of words per list varied from 5 at the A level to 25 at the R level. The majority of the 792 words used in the experiment were words which had been digitized by Texas Instruments and were in the existing contracted list. These words ranged in difficulty from "the" to "mortgage".

For the school year in which the experiment was conducted, the experimental words were removed from the contract lists in order to prevent the experiment from being a review of previously studied words. These words were assigned a difficulty level derived from "The New Iowa Spelling Scale" (Greene and Loomer, 1977) and were ordered from easy to hard. The words were separated into contract lists like those in the original program. They were also put on magnetic disk for use in the machine.

SAMPLE

The experimental subjects consisted of one hundred and sixty-three third and fourth grade students from an upper middle class suburban school. The students were from seven classrooms in one building. Prior to the experiment, four of the classrooms were using the Laidlaw Brothers Publishers spelling series book (1975) and the other three classrooms were using word lists and student-teacher contracts. The students were randomly assigned to treatment groups.

INSTRUMENTATION

At the completion of the experiment a 40 word test was generated for each student by randomly selecting from those words the student had studied. Percent correct on the 40 item test was used as the achievement measure in all analyses.

Throughout the experiment, the teachers were asked to record their time on a daily log sheet. Time was divided into eight categories. Six of the categories pertained to administration or record keeping and two to student services. The entries were charged appropriately to machine or list support activities.

RESULTS

Achievement data collected from the experimental subjects were analysed by treatment group and previous program using analysis of variance procedures. The analysis of immediate achievement data revealed significant differences due to both the previous program F (1,156)=10.039 P<.01 and experimental group F (1,156)=7.356 P<.01. Students whose previous experience had been with spelling books scored significantly higher than those previously using contracted lists. The means were .918 and .865 respectively. Subjects in the machine group scored significantly higher than those in the control group with means of .918 and .873 respectively. The control group students saw on the average 158 words whereas the students on the machine saw an average of 146 words. The range was from 30 to 355 for the control group and from 56 to 222 for the machine group.

On the retention test the data followed the same pattern as for the immediate test. Previous program produced an F (1,154)=4.945, P<.03 whereas for experimental group an F (1,154)= 5.926, P<.02 was calculated. Students whose previous experience had been with the books scored significantly higher than the contracted list classes. The means were .926 and .893 respectively. The mean of the machine group (.930) was significantly higher than the mean of the control group (.894). In no case was there a significant interaction between previous experience and experimental group. A breakdown of these data is shown in Table 1.

Table 1

Immediate Posttest

	Previous Program	
	Book	List
Machine	.931[a](.089)	.901(.075)
List	.905(.084)	.831(.158)

Retention Test

	Previous Program	
	Book	List
Machine	.939(.080)	.918(.085)
List	.914(.070)	.868(.133)

[a] means are percent correct on 40 item spelling test.

The number of words per session did not produce a significant difference in achievement for the 44 students participating in the secondary experiment t(42)=.262,P<.05. A mean of 84.7% was obtained from the students receiving 20 words per session and a mean of 85.8% was obtained from the 30 words-per-session group. The 20 word group saw on the average 92.9 words and the 30 word group saw an average of 132 words. The means for this experiment were lower than the means for the primary experiment because the words were from the upper end of the difficulty range.

Because of start-up problems, teacher time devoted to spelling was summarized by week 1 and weeks 2 through 6. It was further subdivided into management activities and student contact. Teachers spent 5.67 hours on machine spelling and 9.98 hours on list spelling the first week. On the next 5 weeks 6.98 hours were spent on machine spelling and 35.83 on the list program. The big difference was in management activities where teachers spent 4.18 hours with the machine program and 32.4 with the lists.

DISCUSSION

Several aspects of the achievement and retention results merit discussion. It was noted that the students who previously used spelling books scored higher than those using contracted lists. This difference is directly attributable to the way the experimental words were selected, since the words used in the experiment had been removed from the contracted lists and had not been removed from the books. Thus, the students previously using the books were exposed to many words they had studied prior to the time of the experiment. This detracted from the magnitude of the difference between the experimental (machine) and control groups because many students who previously used books topped out on the exam.

Using the same word lists for both achievement and retention tests prompted learning from the achievement test which was reflected in higher scores on the retention test. This also detracted from the true machine effects since the machine group had a greater ceiling effect on the exam. With an achievement test mean of 92% it is very surprising that the scores increased.

In this type of research it is difficult to maintain comparability between the treatment groups. In this experiment due to scheduling difficulties, equipment problems and abscences the machine group saw on the average 12 less words than did the control group. The question arises as to whether this 12 word difference accounted for the achievement test difference. The second experiment, comparing students receiving 20 words per session with those receiving 30 words per session clearly indicates that 12 words out of 158 could not have produced a 4% achievement test difference.

An analysis of teacher time involved with the two systems is quite complicated. It does not include the paraprofessional time which was quite high during the first three days and when equipment failures occurred, so from that standpoint favors the machine. However, the teacher activities attributed to the machine were primarily explaining procedures to students, moving students to the spelling machine or merely watching. These times would be the same whether the entire class used the machine or only half the students were involved. On the other hand, the times attributed to the list program were recording, conferencing and reporting, hence would be doubled if the entire class were involved.

It is estimated that the seven teachers spend 430 hours on the list program per year. The machine, if properly designed, might require 80 hours of paraprofessional time and 60 hours of teacher time. However, as teachers learned to use the machine, the individual attention they gave to students with spelling problems would undoubtedly increase.

CONCLUSION

This project revealed several important issues concerning the future of computers in education. (1) The vast majority of existing research is only peripherically pertinent to many of the key design problems encountered. For example, consider the problem of designing the algorithm to use for selecting words. Prior to computers with randomly accessible word lists, record-keeping capabilities for each student and auditory and visual output, research was constrained by the practicality of the classroom environment. There was no need to know how long to delay before representing a twice missed word because no one could utilize the knowledge if it were available.

(2) The electronics industry does not have a good picture of the needs of the classroom. The curriculum, instructional methodology, and student characteristics are vaguely understood but scheduling and classroom management consideration are appreciated only by teachers and some school administrators.

(3) Strong evidence was collected to indicate that the challenge of building successful instructional technology is presented by the teachers. Success is contingent upon dependability of hardware and the utility of progress reports. Teachers will not sympathize with mechanical problems nor will they condone a sense of insecurity about student progress. At the same time, teachers are not adept at analyzing and communicating their technological needs.

(4) Very cheap terminals with flexible display and both independent and on-line capabilities are essential. The capability for downloading the instructional algorithm for each lesson, from the microcomputer to the terminals, is also necessary. Such a facility would not only permit spelling instruction but would accommodate 70% of the K-8 mathematics objectives and a substantial portion of the language arts objective as well.

Computer managed instruction of the type described above could be realized within the very near future through a cooperative developmental effort between publishers, the electronics industry and educators. Publishers are already producing materials on relatively expensive over the counter hardware. The electronics industry is only two or three years away from the required cheap nonvolatile memories and display technology. Hopefully, educators could define and articulate classroom needs within a relatively short period of time if given technological support. Yet, in spite of the need for improvements in basic education and in spite of the technological feasibility of meeting classroom needs, extensive developmental efforts are very unlikely. These efforts will remain unlikely until it becomes apparent that schools can and will purchase sufficient computer-based systems to make it profitable to divert sales efforts from the home hobbyist to the classroom. Therein lies the tragedy.

Word Processing/Text Editing
Using Microcomputers

Dr. Peter Wetterlind
Mathematics and Computer Science
Department
University of Northern Iowa
Cedar Falls, Iowa

As educators we are well aware of the extensive uses being made of computers. It is widely recognized that computers have become a factor in educational environments, and not just for operating with numbers. This discourse presents a survey of the systems available for processing these character, or non-quantitative data. Acknowledging the limited funds with which we perennially function, these comments shall be directed toward the microcomputer handling of character data. Microcomputers are the small, sole-user oriented machines that may be purchased for less than approximately ten thousand dollars. These data treatment systems are called word processors and/or text editors.

To avoid any misunderstanding and to establish a common base for discussion a distinction shall be made between word processing and text editing. Word processing will refer to the manipulation of characters, words, phrases, and sentences. Text editing, on the other hand, will reference the handling of whole documents which are created from these elementary constructs. There are two general types of word processing packages. In computer jargon these are called line oriented and page, or screen oriented systems. After a brief comparison of the processes used in each of these two systems, two of the varieties of text editing packages shall be viewed.

There are three basic word processing operations. These are the 1) changing or replacing of single characters, 2) inserting additional materials into an existing text file, and 3) deleting material from a file. When using a line oriented word processor the computer quite simply responds to user supplied commands. The screen does not, typically, reflect the results of such actions. That is, the computer assumes the user to know the contents

132

of the current line, and also to trust the computer to unerringly perform the designated task. The user is allowed to move an imaginary pointer around in the text file by entering such elementary commands as A10 - which means to ADVANCE the pointer 10 lines, B6 - meaning to BACK-UP the pointer by 6 lines, R19 - to shift the pointer to the RIGHT by 19 spaces, and L3 -shift LEFT by 3 spaces. When the pointer has been properly positioned, the user can change characters by entering, for example, Cx - which means CHANGE to "x" the character being pointed to, or insert text by entering a command such as Ithe new - meaning to INSERT "the new" into the file at the current pointer position, or delete text with a command like D16 - meaning to DELETE the next 16 characters starting at the pointer. A word processing system such as this is relatively easy to implement on a computer and requires a minimum of machine resources. Hence, it is inexpensive and currently available for almost any microcomputer. Its primary deficiency is the degree of "intimacy" required between the user and the text file being processed. On the other hand, the page oriented word processing systems are more friendly toward the user, more elaborate, and more costly. The same three basic operations may be performed. In this system the user is offered a copy of part of the text file on the screen. A pointer can still be moved around in the file, but there is also a physical image of the pointer moving around on the screen. Pointer motion is usually controlled by some terminal keys with left, right, up, and down-pointing arrows. Adding these arrow keys to the system and providing their functions contributes to the cost of such word processing packages. After positioning the pointer, characters may be changed with this system simply by overprinting. The screen contents are reflected in a work space within the computer. Material may be inserted or deleted by using a special command form such as depressing the control (CTRL) key and simultaneously depressing either the I or D key. As in the other system, inserting material into a file causes the trailing contents of the line to be right-shifted to make room for the new material. It is the user's responsibility to insure that lines are not too long for the computer to store. Excess material is usually discarded. The user must also accept the task of filling the lines from which text has been deleted. In both of these systems the computer typically stores the original version of the file as a back up copy, and copies the updated version into the space formerly occupied by the original. Files in these forms may simply be printed, or they may be subjected to a form of text processing prior to printing.

Text editing packages generally include two functions. The text file may be reformatted prior to its printing, or there may be a merging of two files resulting in the creation and/or printing of a third file. One such commercially available version of a text formatter accepts a text file, scans it for specific format instructions, and produces a revised form of the file which has been created by carrying out the imbedded instructions. For example, the command $C in the first two positions of a line might be used to cause the next line of text to be centered with respect to the rest of the text. Indenting a line 7 positions could be accomplished by typing $I7 in the first three positions of a line and on the next line typing the text line. In a system such as this where the computer considers specific lines and characters, it is entirely reasonable to expect it to also scan over successive text lines and use the spaces as delimiters for parts of lines to append onto preceding lines so that a consistent right margin for the text may be achieved. In cases where the lines are only slightly abbreviated the computer could simply add extra spaces between words to effect the desired margin. The user could type $J at the beginning of the text file to initiate this action. Variety two of available text formatting packages also takes advantage of the computer's "intelligence". If the computer is clever enough to concatenate words onto nearly full lines, then there is no real need for the user to ever be concerned with line boundaries. This system allows the user to enter text continuously, and only when a new paragraph is desired is the "return" key depressed. Of course, such functions as centering a text line must be manually accomplished, but the convenience of never having to slow down for the end of a text line outweighs the infrequent inconvenience of centering a text line. In this system the original text file is not ready for printing until the computer can arrange it. Quite often this type of system does not allow for a constant right margin in the printed file.

The merging of two files is another function available in microcomputer text processing packages. The general application of this function is to create a base file which includes special delimiters, and a merge file containing elements to be substituted for the delimiters as each new copy of the base file is printed. For example, a base file could be a form letter, while the merge file could be the names and addresses of all people to whom the letter is being sent. A simple MERGE F1, F2 could be used to print file F1 successively, and with each iteration incorporate a different set of elements from file F2. Naturally this process would include the formatting of the newly created copy of F1 so the margins would be ordered and the base file would be tailor-made for each recipient.

All of these package functions are currently available, and priced within the budgets of most households as well as schools. The prospective buyer has only to choose the desired features and purchase the appropriate computer and word/processing text editing system. We, as educators, must be aware of these conveniences and be prepared both to use them and to help others to do the same.

COMPUTER LITERACY TRAINING NEEDS OF INSERVICE AND PRESERVICE TEACHERS: IMPLICATIONS FOR TEACHER EDUCATION PROGRAMS

Although computer literacy training is being touted as the next crisis in American education (Molnar, 1978), few colleges of education are presently offering courses that adequately develop a teacher's ability to use the computer effectively in the classroom (Milner, 1979, Podemski, 1981). Machinery has clearly defined competencies that they consider to be "...the scope and sequence of teacher training needed to integrate computing into the schools (Taylor, Poirot, Powell, 1980)." Whether or not teacher education programs will implement these competencies remains to be seen. Their adoption depends on several factors that may be insurmountable: the change in the traditional focus of teacher training programs; changes in attitudes toward the use of instructional technology in general and the use of computers in particular; and the willingness of teacher educators to retrain so they have the knowledge to teach others about the educational applications of the computer. In addition, the successful development of computer literacy training programs in colleges of education will depend on how much consideration is given to the problems that have occurred in the past with the adoption of other instructional innovations (television); how realistic the computing competencies are in relationship to what a teacher really does in the classroom and how well the instructional strategies used to train teachers reflect the teacher's classroom needs. This paper will discuss these issues as well as discuss ways that the University of Texas at Austin is approaching the problem of training preservice and inservice teachers to be computer literate.

In a recent article Podemski (1981) states that the primary responsibility of teacher education programs has been to provide preservice teachers with an area of concentration or specialization such as mathematics and to give them a course work in generic teaching skills.
Computer skills have not been seen as critical part of teacher training. Indeed on the preservice level there is little exposure to the use of any instructional technology. Students may take a basic media course that serves as an overview to the selection and use of common classroom media (film, transparencies, slides etc.), however, this course is often seen as being supplimental to their course of study. Instructional technology is not seen as an essential part of classroom teaching and preservice teachers rarely have a chance to actually see it used successfully in the college classroom.

Teacher educators have not yet moved into the field of technology, therefore, they are not producing teachers who have a knowledge of technology in general much less computers (Stolurow, 1980). This neglect on the part of teacher trainers may have another ramification when it comes to the inservice teacher. Because beginning teachers often model their behavior based on their observations of other more experienced teachers. They rely heavily on what they have been taught in college. If beginning teachers only see and have experiences with traditional methods and tools, they are not likely to use or have an interest in new technology. Old teaching patterns are difficult to change no matter what level: elementary school or college.

Whereas the majority of colleges of education have remained unresponsive in teaching about computers, business has not. Computer stores are offering training courses for educators and as a result are selling a good number of computers to school systems. However, the kind of training done by computer salespeople is not necessarily appropriate to the classroom situation. A knowledge of the BASIC language may not provide teachers with skills needed to use computers effectively in the classroom.

The invasion of computers will not only effect student work, but it is also going to change the role of the teacher and the nature of his or her work. As teachers use the computer to deliver instruction, they will become managers of learning. Essential skills will include a knowledge of educational applications and an ability to match student to the particular program as well as monitor the student's progress. In the past teachers have been the central figure in the delivery of instruction, but when the computer is used as an object of instruction (students learn to control the computer), the teacher's role changes to that of a facilitator. If schools wish to implement this innovation, the issue of changing teacher role must be addressed. Other innovations such as programmed instruction or instructional television have fallen to the wayside because this issue has been ignored. A change in teacher role must be addressed on both the inservice and preservice level. If colleges of education are at all sensitive to classroom needs, they should begin to design teaching methods courses to reflect these changes. Unfortunately, it is not unusual to have teachers complain about their course work having little applicability to the classroom experience.

Efforts to expose preservice teachers to computer technology will require some staff development on the part of teacher training institutions. In other words, college faculty need to be made computer and technologically literate. College professors need to know what a computer can do in education. They need guidance so they can begin to teach about and use technology in their own course work. In addition, teacher educators need to know that there is a grassroots movement urging that computer literacy be a part of K-12 curriculum. This effort to infuse technology into the preservice teacher's curriculum will take a financial and programmatic commitment by the college administration. It will require support staff, equipment and software for it is difficult to teach about computer use when one is not available for demonstration.

There are several ways that computer literacy has been introduced into the teacher education curriculum at the University of Texas. First, the computer could become a component of the current instructional technology or basic media course. This has been done at the University of Texas. Microcomputer technology is studied and students are given the option to design an instructional program as one of their three media projects. Students are provided with a brief overview of educational applications and given a simple introduction to BASIC.

The second method is to add an additional course in computer literacy. This spring semester the University of Texas offered its first preservice course (a graduate level course has been taught for two years). The course content covers the following: the history of computing, computer components and terminology, BASIC programming, computer literacy curriculum K-12, computer literacy teaching materials and resources, educational applications, hardware and software selection, computer careers, societal impact, and future technology.

The third method is to redesign the current methods courses to include a computer literacy component and to begin to teach methods for facilitating and managing learning. The best the University has been able to do is to include the microcomputer as a component to a basic media skills course that is taught in conjunction with the methods course. Although it is a start, this still leaves the students without adequate training in computer use.

Other ways to help infuse computers into the curriculum is to provide for workshops for faculty and students. The Learning Resources Center in the College of Education at U.T. provides regularly scheduled as well as faculty requested orientations to the microcomputer laboratory. These orientations provide the participant with hands on experience and a brief overview of the educational software packages that are available for faculty and student use. The Learning Resources Center is a good example of both staff and materials support of a technology. Not only does it house the computer equipment and software, it maintains a staff of teaching assistants who give orientations and help in the microcomputer lab. The lab has also been used for university wide workshops sponsored by Center for Teaching Effectiveness. Continuing education efforts and extension courses are also important. At U.T. these efforts are not as great because Texas has regional education service centers that do inservice training. Some cooperative efforts have been made with the service center for it is an excellent way for the University to keep in touch with inservice teacher training needs.

Although the University of Texas has done much in the area of computer literacy, it still will not be enough to ensure that all teachers leaving our institution have a clear understanding of the computer as an instructional technology. The courses are not required and unless the certification requirements for teachers in Texas change it is not likely to be made a requirement. At this time, no faculty members other than those in Instructional Technology are using the microcomputers as an instructional tool. Computers may be taught as an additional component of a course, but unless the technology is infused throughout the curriculum, we cannot expect preservice teachers to fully understand its educational use. Nor can we expect them to adopt the technology as a teaching tool. But there is always hope, because somehow this technology is creeping slowly into every facet of our lives. Teacher Educators will inevitably feel the pressure and lure of the computer.

Molnar, A. "The Next Great Crisis in American Education: Computer Literacy" AEDS Journal, Vol. 12, No. 1, Fall 1978.

Milner, S. "An Analysis of Computer Education Needs for K-12 Teachers" National Educational Computing Conference Proceedings, Iowa City, 1979.

Podemski, S. "Computer Technology and Teacher Education" Journal of Teacher Education, January, 1981, Vol 32, No. 1.

Stolurow, L. M. Editorial, T.H.E. Journal, 1980, 7 (2).

Taylor, Poirot, and Powell "Computing Competencies for School Teachers." National Educational Computing Conference, Norfolk, 1980.

CALCULUS TUTORIAL SYSTEM

Nicholas Arganski
Department of Mathematics
and
Computer Science
Ste.Anne de Bellevue, Quebec
H9X 3L9

The Calculus Tutorial System (CTS) consists of five main subsystems: The Interactive Derivative Solving System (IDSS), the Interactive Integral Solving System (IISS), the John Abbott College Individual Supplementary Calculus Instruction (JACISCI), the Computer Generated Individual Problem Sets (CGIPS) and GRAFIT - a practical graphing program. Of these, the IDSS, the IISS, the NEWPROB of JACISCI and the CGIPS constitute unique and innovative programs which originated at John Abbott College. Due to space limitations only the IDSS is discussed at some length in this paper. The system is written in BASIC and is presently executable on an H.P. 3000.

Introduction

The IDSS and IISS (not completed) enable a first year student of Calculus to find derivatives and integrals (with explanations) of any algebraic or transcendental function that he/she might come across. The NEWPROB of the JACISCI enables any non-programming teacher to prepare a lesson (a series of problem files) on any topic of his choice in an easy and effective manner. The JACISCI processes students' on-line lessons prepared with the help of the NEWPROB. It provides immediate feedback and grading of students and keeps extensive records. Finally, the CGIPS offers five different approaches to computer testing with extensive records and statistical results.

IDSS

The original idea occurred to two of our students, Rick Martin and Scott Darlington, some seven years ago. Rick and Scott realized that since by its very definition the operation of differentiation is recursive in nature - a rule for differentiating a function u(v) can always be found in the form:

$$\begin{aligned} d/dx\ u(v(x)) &= u'(v(x))\ dv/dx \\ &= f(u(v(x)))dv/dx\ , \end{aligned}$$

they decided to use a recursive algorithm to do the differentiation. The utility of the design is due to the fact that with the use of the recursive algorithm separate modules can be written for the differentiation of various functions u - each being completely independent since we need to know nothing about the argument v to form the derivative of u(v). We need merely pass v in a recursive call to the system to find dv/dx.

The only thing left to determine at this stage is the interface between the modules. For this purpose we require a common block of shared variables which will be set up just prior to each call. We wish basically to pass a function F$ to any module and have it pass back the derivative string D$. We will also provide a flag variable F to indicate the status of the returning program - it will be set non-zero if an unrecoverable error is encountered.

Once we have an algorithm available to determine which modules we require to differentiate our function u(v), the design of the specific modules is relatively simple. Thus the basic function of the interpreter section is to take a function string given as input, identify the terms, and call the appropriate modules to differentiate it. It should then return the derivative of the function as output. This task is easier if we make the initial assumption that our function string has already been broken up into terms in a sum or difference. Thus the interpreter section of our system can be conveniently broken up into two separate stages:

i) Splitting the function into terms in a sum or difference;
ii) Taking the simple terms from i) above, identifying them, and calling the appropriate modules to differentiate them.

These tasks were allocated to the modules IDSSPLIT and IDSSPRTR (for interPReTeR) respectively, see flowchart for MODULE RELATIONS.

To explain this division of labour more clearly, let us take the function string "2x - sin(y)" as an example.The string would first be split into three parts by IDSSPLIT: the infix operator "-", and its arguments "2x" and "sin(y)" which would be passed to IDSSPRTR in succession. IDSSPRTR would identify the first as power rule and invoke the power rule differentiation module. Next, the trigonometric function differentiation module would be called by IDSSPRTR to take care of "sin(y)". These derivatives would be passed back to IDSSPRTR and to IDSSPLIT and the infix operator "-" applied to them, giving "2-cos(y)(y')". Note that the trig function module did not need to know how to differentiate the argument "y" to differentiate the function sin(y), it merely invokes IDSSPLIT to differentiate it, using the chain rule. The only module which does not necessarily invoke the interpreter is the power rule module. Thus the power rule module will be called for every function to be differentiated at least once - to allow for termination of the system.

Thus the task of the IDSSPLIT module is relatively simple. All it has to do is to identify 'break points' in a function - plus or minus operators, where the function can be split - and remember the operator while passing the argument to the IDSSPRTR module. This is made relatively easy by the fact that a plus or minus symbol external to any parentheses is a valid break point. The only other task given to the module is to resolve the infix operator with any preceding sign in the derivative of the second argument, thus, for example, producing "2-2x" from "2", "+", and "-2x".

The task of the IDSSPRTR module is much more complex than that of IDSSPLIT. It has to identify any of the following forms:

1. power rule (au**b)
2. chain rule
3. a**u
4. keyword (argument) such as sin(xy)
5. product rule. Must also locate all terms in the product and how many there are.
6. quotient rule. Must locate all terms in both the numerator and denominator of the quotient.

These tasks are performed in conjunction with modules 02 to 14 (see Module Relations).

The operations performed by the rest of the modules in the system are as follows:

DIALOG: Introduction to the system via dialogue; user receives instructions on how to use the system.

IDSS00: Program/syntax control. Checks for correct format, balanced brackets, etc.

IDSPAUSE: Stops the program during dialogue so that the user can catch up. The module prints a colon (:) and waits for the user's response. If just a carriage return is entered, the module ends. If STOP is entered, the flag F is set to one and the module ends. If NEW is entered (the user wants to enter a new function) the flag F is set to two and the module ends. If QUIET is entered (user wants no explanation - only the answer) the flag F(1) is set to true and the module ends. If COMM is entered, the module allows the user to enter a comment. The comment is stored on the file IDSSCOMM along with the

141

student name, ID #, date and time. Once the comment is completed, the module ends.

IDSSDATA: General purpose data.
IDSS12: Bracket removal.
IDSSIMPL: Simplification of result. Eliminates terms equal to zero and parts of terms equal to one.
IDSSDIV: Division of display module. Breaks a quotient result into numerator and denominator and displays in a more easy to read format.
IDSSPLIC1 to 4: Implicit Different'n.
IDSS02 - product rule.
IDSS03 - quotient rule.
IDSS04 - power rule.
IDSS05 - chain rule.
IDSS06 - trig.
IDSS07 - inverse trig.
IDSS08 - exponential.
IDSS09 - natural log.
IDSS10 - a**u
IDSS11 - log base 10 .
IDSS13 - hyperbolics.
IDSS14 - inverse hyperbolics.

The system also has a USERFILE (users are addressed by first names) and a LOGFILE which provides an extensive record of each session.

IISS - Many of the ideas (and programs) for this system were taken from the IDSS. The system is composed of 31 program modules written in BASIC for HP3000. Since the process of integration in some cases involves differentiation (integration by substitution), one of the IISS modules has been written to act as an interface between IISS and IDSS. Through this interface the IISS can differentiate any function it needs to. Presently the IISS can integrate functions of the form Ax^n, simple trig, hyperbolic and inverse trig functions, and does integration by substitution with two factors for the following cases: (i) If one factor is the derivative of the other. (ii) If one factor is the derivative of the argument of the other. (iii) If one factor is the derivative of the base of a power. (iv) If for any one of the three above the derivative differs only by a constant factor. The system also has utility modules and files similar to those of IDSS.

JACISC - This system was derived from the ISCI originally developed at the University of Minnesota by the Consulting Group on Instructional Design, under the leadership of Dr. Leonard Shapiro. The original version was written in FORTRAN and consisted of some half a dozen programs. The JACISCI consists of over forty programs and is written in BASIC.

CGIPS - Developed by Dr. G. Welch and Dr. C. Robertson of JAC's Department of Chemistry, the system was adopted by CTS in 1978. It prepares and processes tests of five types: (i) multiple choice; (ii) numeric problems (algorithmic); (iii) multiple choice problems (hybrid of types (i) and (ii)); (iv) essay questions; (v) general purpose (multiple options). The CTS-CGIPS consists of about 2000 problems, most being type (i).

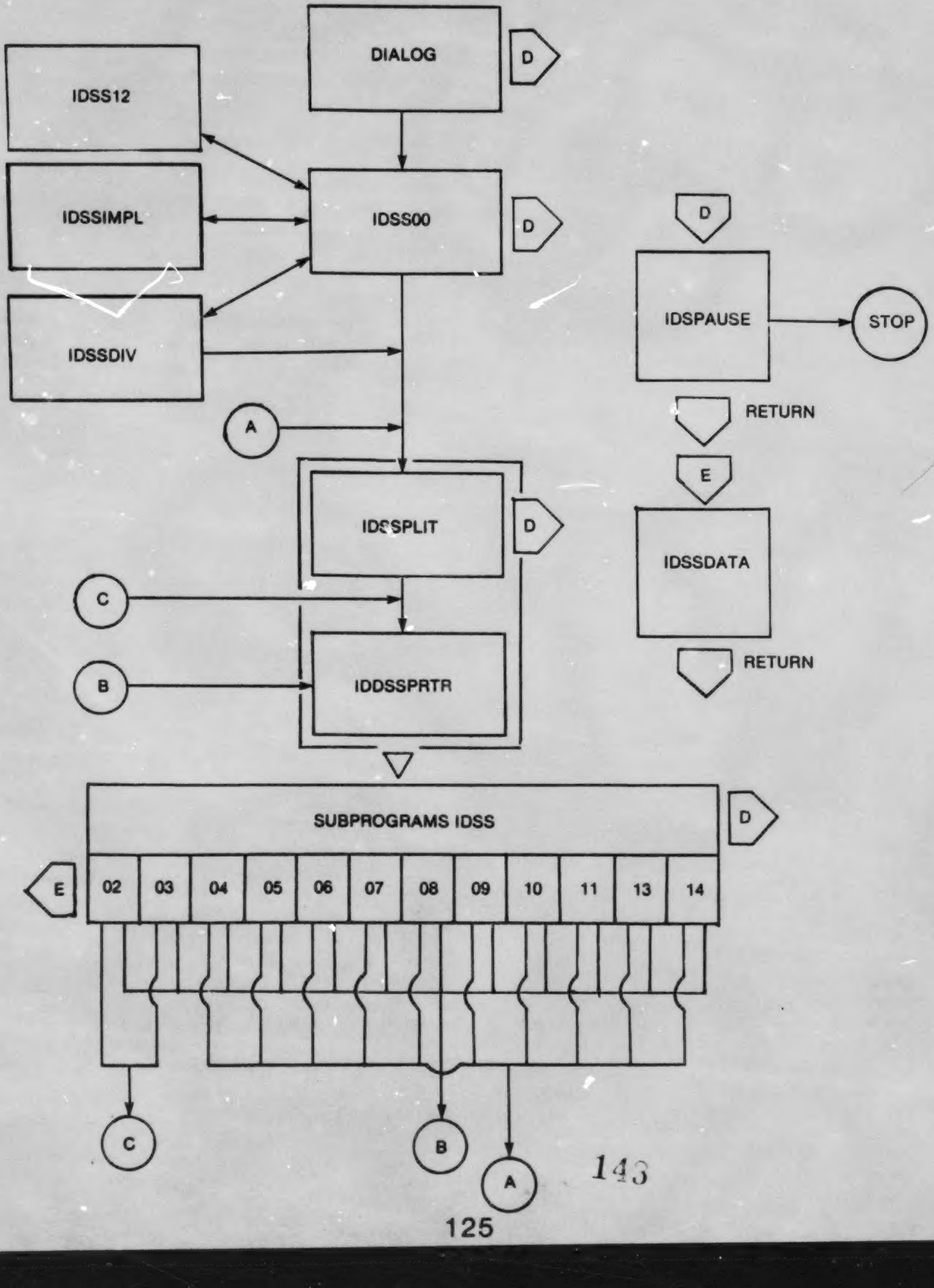
GRAFIT - Mainly based on work done by R.M. Palay of Washenaw Community College, the JAC version has been developed by Robert Mearns, my partner in the project, using a Diablo.

STATISTICAL ANALYSIS - In the fall semester of 1980 the CTS was tested using a t-test. The control group had 57 students and the experimental - 28. The H_0 (there is no difference between the groups) was rejected at .1 level of significance in favour of the experimental group.

ACKNOWLEDGEMENTS

In chronological order, the JAC student contributions to the project are: Rick Martin, Scott Darlington (lion's share), Glenn Connery, Ken Robertson, Larry Barth, Geoff Hendren, Ronald Shutler (lion's share), Irving Witte, Colin Cameron, Loulou Fitzgerald,David MacKie, Andy Gauthier, Ralph Kroll, and Robert Parnigani. Without their devotion and hard labour this project would never have materialized.

MODULE RELATIONS


DIALOG
D
IDSS12
IDSSIMPL
IDSS00
D
IDSSDIV
D
IDSPAUSE
STOP
RETURN
E
IDSSDATA
RETURN
A
IDSSPLIT
D
C
B
IDDSSPRTR
SUBPROGRAMS IDSS
D
E
02
03
04
05
06
07
08
09
10
11
13
14
C
B
A

143

HIGHER EDUCATION'S RESPONSIBILITY TO JOB TURNOVER IN COMPUTER-RELATED JOBS

M. Gene Bailey and David Black
Western Carolina University

Statistics indicate that the job turnover rate of people in computer-related fields is one of the highest of all professions. It has been estimated that one out of every three people in the data processing field will change jobs this year.[1] This presents a problem for companies that are highly dependent upon data processing. Turnover is expensive because of time and money lost not only in training a new employee, but also, in trying to recruit in a market that has a scarcity of qualified personnel.

What are the factors involved that cause job dissatisfaction? Evidence indicates that, although salary is often mentioned as the reason for leaving, it is often only incidental. [2] Among the areas, then, which must be considered are job expectation and job preparedness.

Universities must address directly the problem of turnover in two ways. First, they must provide the proper academic courses and training. Second, they must prepare the student's attitudes so he will be aware of existing problems and will not be over enthusiastic. Fresh from schools, new programmers prefer--and widely expect--to tackle sophisticated software development assignments soon after they start their employment. However, managers cannot give bright entry level programmers immediate development assignments because that would require transfers of veteran programmers from vital and hard-to-fill roles to temporary duties or trainers of new hires.[3]

A survey conducted at the National ACM meeting in St. Louis, 1981, provided some interesting results. Students interviewing for jobs were asked what they thought employers wanted from graduates. First among their responses was "good grades." This same response, however, ranked at most second, when the same question was asked of employers. In the latter case, communication skills were among the most important. It is essential that the data processing staff be able to communicate with the rest of the organization. This includes co-workers, users of the computer system, and other professionals that require their expertise. There were other examples of discrepancy that were stated--all verifying that there exists a "communication gap" between industry and graduates of information systems and computer science. It is the responsibility of higher education to help the student to bridge this gap.

Universities and colleges have a responsibility they must assume in teaching and training management to work with the type of individual who enters a computer field. Data processing personnel are unique. The type of work they do and the way their minds work are very different from other people in the organization.[4] For these reasons, management must be especially sensitive to and aware of their needs.

What are these needs? How can management, once made aware of them, deal with them? Job challenge seems a particular failure in data processing and a major cause of turnover. People want enriched jobs, yet both top management and data processing executives have tended to confine their people to strictly technical roles.[5] Lack of recognition is a leading factor in the loss of morale. The rate of turnover can be halted or reduced in an organization by demonstrating that management cares. Data processors want desirable work atmospheres, a feeling of accomplishment, opportunities to advance and recognition.[4]

This paper addresses two of these major issues: what can higher education do directly to better prepare students for their respective careers, and what can it do indirectly to prepare management to cope with the data processing individual?

Students should have to review current journals to keep abreast of new trends and ideas. They should also be encouraged to stay constantly aware of their marketable skills. The particular skills necessary will depend upon the specific technological paths that data processing and the computer industry are taking at a given time.

For a university's Computer Information Systems Department, this means that they must continually reevaluate, update, and change curriculum. It is important to provide a realistic educational experience in the design and implementation of computer software systems. Unfortunately, the typical academic courses involve short program assignments done on an individual basis.

The major difference between academic and production programming requirements include:[6]
1) The problems to be solved in industry are often more complex and of a larger scale than those encountered in homework assignments. Furthermore, academic courses place a heavy emphasis on systems programming jobs dealing with compilers and operating systems, while production work consists of both system software and application software.
2) In contrast to the individual work that is invariably stressed in the academic environment, production programming of software is often developed by programming teams under constraints in both time and resources.
3) In the traditional academic setting, students seldom have the opportunity to develop software requirements and detailed design specification.
4) In industry, large volumes of software documentation are required and must be produced to meet stringent standards. Western Carolina University has approached this particular problem by instituting a new one-semester course designed to follow a software design course taught the previous semester. Students develop system analysis skills during the fall term by studying and analyzing case studies of existing systems.

At the beginning of the second term, the students are divided into teams of six to eight. Each team is assigned two team leaders whose primary function is to assume the role of "system analyst." Projects solicited from industries, local businesses, and even government are assigned to individual teams. Under faculty supervision, they meet with their respective clients for a short period of time to develop the goals and criteria for achieving them. Once the projects are into design stage, the communication with the client is through the team leaders.

The primary objective of the course is to provide students an opportunity to work on problems that are of the magnitude and complexity of those found in the "real world." In addition, valuable skills such as software project management and teamwork can be easily taught. These are normally difficult, if not impossible, to teach effectively in the classroom.

Western Carolina University is extremely active in the cooperative education program. Students are encouraged to spend two non-consecutive semesters employed by an organization that will allow them to work in their area of interest. Thus, students have an opportunity to sample the "real-world" version of their major. The shock and often disappointment or disillusionment normally encountered upon entering the job is greatly reduced.

Cooperative education could

benefit not only students, but the faculty, and thus, the university as well. Companies are becoming very receptive to the idea of hiring faculty on a semester or yearly basis. For the company, the expertise and research experience of the faculty member is obtained. Faculty will be made aware of current status and problems of "real-world" programming and will be able to adapt methods and theories in their classrooms to better prepare students for actual conditions. Western Carolina University is currently investigating the possibility of this "all-university" type of co-op program.

Students graduating from computer information systems and computer science programs are greatly lacking in communications skills. As mentioned earlier, it is absolutely essential that the data processing staff be able to communicate with the rest of the organization. This includes co-workers, users of the computer system, and other professionals that require their expertise. This communication will be in both written and oral form. Reports must be produced that are clear and concise-objectives clearly stated; methodology, results, and conclusions clearly defined.

Universities, especially faculty in the technical areas, are prone to de-emphasize this most important skill. As a result, the graduate, though highly trained in his specialty area, is frustrated by these "extraneous" demands placed upon him in his new surroundings. Schools that have omitted courses that assist the student in developing communication skills must reinstitute them into their programs. Those that have them must emphasize them.

To be able to communicate with a user requires more than being able to write and speak fluently; the particular speciality of the user is important. Highly technical graduates are confronted with problems they could easily solve if they were familiar with the subject. For example, a graduate with no business courses to his credit would have a more difficult time developing software for an accounting firm than a graduate with the same computer skills who had taken business courses.

Higher education's responsibility to students studying for careers in computer-related jobs extends to include the responsibility for educating the people who will manage them. Growth of the computer industry has been explosive--graphed, the line would be almost vertical. Managers lacking technical expertise are supervising employees who are highly technical. For these managers, workshops could be offered which would focus on the problems of computer professionals. For a long-term solution, education needs to examine curricula in management courses to insure that future managers will be familiar with the type of individual working with computers and his special problems.

Western Carolina University has taken some steps in this latter direction by changing the emphasis of some of the business core courses. In particular, all students in the School of Business are required to take a management information systems class in which not only the management of information systems is discussed, but the management of information systems personnel is discussed as well. All departments in the School of Business are actively involved in computer usage--giving future users and managers the opportunity to interact with programming and systems personnel at the university's computer center.

The turnover rate problem can be reduced through training at college level to make students aware of "real-world" situations. Management should have workshops to update knowledge of techniques. Both areas need to improve communication skills--a feat to be accomplished with tactfulness and patience. WCU is one university making the effort to close the gaps through training and workshops. Higher educational institutions as well as businesses should focus on these areas to emphasize efforts toward reducing turnover rate.

BIBLIOGRAPHY

[1] Steven Stibbens, "People: The Cost of Finding, Hiring and Training," *Infosystems*, January 1981, Vol. 28, No. 1, pg. 46.

[2] Merrill Cherlin, "Burnout: Victims and Avoidances," *Datamation*, Ki; u 8, 1981, pg. 92.

[3] Brad Schultz, "Challenge, Recognition Lift DP Morale, Survey Finds," *Computerworld*, August 10, 1981, pg. 1.

[4] Bob Johnson, "To Halt Turnover, Show Them You Care," *Computerworld*, September 7, 1981, pg. 15.

[5] Janet Crane, "Personnel Trends: A Look at the Insurance Industry," *Datamation*, July 8, 1981, pg. 76.

[6] Stauros N. Busenberg and Wing C. Tam, "An Academic Program Providing Realistic Training In Software Engineering," *Communications of the ACM*, June 1979, Vol. 22, No. 6, pg. 341.

MICROCOMPUTERS IN THE GIFTED/ TALENTED PROGRAM

Mary Berry

Merry Bee Communications

Meeting unique individual needs is the primary difficulty in mainstreaming any child with wide deviations from the norm - whether visually impaired, mentally handicapped, or gifted. Most schools have chosen enrichment rather than acceleration or segregated classes as the framework for their gifted programs. The time to plan enrichment activities, locate materials and give the individual instructional guidance for a single gifted student is a burden to the teacher whose day is full with the preparation, instruction and evaluation of scheduled classes.

The computer can be used as a tool for enrichment. The teacher can prescribe a program letting the computer become the individual's instructor. Educational software (especially earlier efforts) is often criticized as being too difficult for the intended audience. For example, a math drill with the reading level too high for the grade level where the concept is introduced or a social studies simulation with too much information shown too quickly to assimilate the import of the facts. These materials would be a solution rather than a problem for the gifted student who needs his grade level content presented at the advanced level of his learning ability.

The teacher can develop a program to provide individualized enrichment. Since writing a program is seldom quickly accomplished, this approach would seem to be adding to the burden of the teacher. However, a caution in working with gifted children suggests how program development can be a viable alternative. That caution is, "Don't do too much for them." Don't just ask the gifted child to learn the material, rather let him do the experiment or gather the facts to develop the concept. But don't just ask the gifted child to fact find and conceptualize, rather let him design the experiment/search as well as carry it out as well as generalize the principles/learning.

Development doesn't have to be entirely original. The gifted student usually responds to the challenge to adapt or add to an existing program. Program listings in periodicals are often good starters. For example, one Creative Computing article contained a listing for a graphics Christmas tree with blinking lights for entry in hi-res on an Apple. How to add other decorations to the tree is an interesting challenge. Students can adapt certain games or simulations to give the advantage to a certain character or to make it easier for the "good guys" to come out ahead.

One adaptation to the curriculum for the gifted student is the incorporation of experiences designed to develop thinking skills. Various systems to train teachers to do this within the setting of the regular classroom dialog are available. Specialized across discipline materials may be purchased.

The computer can be used as a tool to develop thinking skills. Specific assignments in programming can be constructed which will develop skills in both inductive and deductive thinking, and problem solving techniques. These can vary from graphing a picture for computer display to creating a program to carry out some function in the student's area of interest (example, figuring baseball statistics for the school team). They may utilize one of the special child languages such as TI Logo, Atari PILOT or Smalltalk.

While some keepers of the computer outlaw games in the school setting, the logic development possibilities of some games should not be overlooked. Again there is a great variation in the sophistication and level of available programs. These can vary from a simple "Bagels" to "Othello" to an adventure type game in which success depends on a strategy of exploration and remembering what has been discovered.

Another type of challenge is presented by the ability of the computer to recreate real life situations by means of simulations. The student is led to do research in order to produce his own accurate simulations. He can call on thinking skills appropriate to his level to analyze cause/effect relationships or trends and to generate formulae to represent these in the simulation.

Gifted/talented students do not come uniformly perfect. Each has his own areas of ability and interest. The computer can be used as a tool for the student to build on his own areas of strength. This is of particular importance when the area of strength is mathematics. The mathematical talented younger student is a problem for the classroom teacher who doesn't want to merely advance assignments on into books used by the junior or senior high but doesn't want to assign meaningless drill. Some horizontal enrichment can be provided by using computer programs in the areas of measurement (metric especially), geometry and word problems. The child can combine his knowledge of grade level operations with his developing programming skills to create games and word problems for his classmates.

Another method for building on areas of strength is the pairing of the student with a mentor. Locating a willing mentor and arranging time together can be a prohibitive problem. The student can take materials in the area of his strength and explore beyond what any person available in his immediate environment can provide directly. This could be especially helpful in the areas of music with computer generater and art with graphics packages on the market.

A corollary to having areas of strength is that there are other areas of weakness. Gifted students typically attempt to use their areas of strength to compensate for these whenever they can't just be ignored. For example, the primary child for whom reading is difficult can do well on tests by listening attentively to class discussion.

Instead of pretending that these areas of weakness do not exist, the computer can be used as a tool for overcoming deficits. The appeal of the machine is the major aspect in the use of the computer. The gifted student with weaknesses in basic skill areas which lie outside his interests can be captured for needed drill by the motivational influence of the computer.

Occasionally a student is hampered by a deficiency other than a weakness in a content area. A gifted student may fear to put his self-concept in jeopardy by risk taking. Working in isolation with the computer can be an excellent experience for the child who is afraid to orally volunteer his ideas in discussion for fear of being wrong. The lessoned anxiety of making his mistakes in private with a patient computer should increase acceptance of risk taking.

Areas of deficiency can also be due to physical factors. Special adaptations in hardware and software can help the disabled gifted child to realize his potential. Many examples of projects and organizations with on-going or experimental programs geared for individuals with various disabilities are appearing in the literature.

Whatever provisions are made for educating the gifted/talented student, management is a key factor in the program's success. Here again, the computer can be an aid, although the specific management use will depend on the educational model adopted. Some general techniques may serve to suggest specific ways in which the computer can be used as a tool for management in the gifted program. A computer can assist whenever any type of flexible scheduling is a component of the program. The computer can assist in the analysis of a range of sociometric type data important in considering the guidance needs of the student. Data management programming can assist in preparing a student profile summarizing comparative areas of growth and analyzing deficiencies.

Microcomputers are a definite aid in

providing for the needs of the gifted/talented student. However, the educator using the computer with the gifted program should be aware of possible problems. Opportunities for social growth are limited by time spent in isolation with the computer. Gifted students at the intermediate level frequently face some problems in self-identity and group adjustment as the age is reached at which peers begin to take on an increased importance. The computer can become an escape from having to deal with these problems. The mechanism for appropriate guidance should be available to deal with ramifications.

Another potential problem area is created by the extensive use of similar programming. Linear or "right answer" programs limit creative thinking. Consideration should be given to the cumulative effect of drill programs on the thinking skills.

The need to fit the educational program to the child rather than the child to the program is magnified with computer involvement.

EFFECTIVENESS OF COMPUTER-BASED INSTRUCTION IN JOB CORPS

Daniel M. Geller, Ph.D.
TEAM Associates, Inc.

INTRODUCTION

In recent years, interest in the effectiveness of computer-based education in improving the teaching of basic educational skills has been growing. Complete computer-based programs include both computer-assisted instruction (CAI), i.e., computer presentation of instructional, diagnostic, prescriptive, and testing materials in an individualized, self-paced format, and computer-managed instruction (CMI), i.e., computer management of individualized, self-paced instructional and testing materials not provided by the computer. While many CAI, CMI, and CAI/CMI programs have been implemented in a variety of educational settings, few have been systematized over a proven matrix of paper and pencil materials covering a wide scope of basic skills. Nor have many included provisions for ongoing evaluation of program effectiveness using an experimental design involving a large sample of disadvantaged youth. One comprehensive implementation and evaluation of CAI/CMI has been undertaken by the Job Corps, as part of its Educational Improvement Effort. The preliminary findings of this ongoing effort are described here.

Job Corps is a federally funded residential program for disadvantaged, out of school youth between the ages of 16 and 22. The program is operated primarily by private sector contractors. The average length of stay for a corpsmember is slightly more than six months. Although the goal of Job Corps is to provide youth with marketable job skills, the Department of Labor has recognized the importance of providing complementary basic educational skills. In an effort to upgrade the educational programs available, Job Corps authorized the establishment of the Educational Improvement Effort (EIE) under contract to TEAM Associates, Inc., a Washington, D.C. based firm. EIE includes the development of new materials, improvement of existing paper and pencil materials, and experimental/control comparisons between program models. These improved programs were implemented and tested beginning in 1979 at 17 of the more than 100 Job Corps Centers. Data have been collected on the progress of nearly 8,000 participating corpsmembers to date.

Program Design

One of the most extensive educational improvement programs has been that conducted in CAI/CMI at the Gary Job Corps Center in San Marcos, Texas. (With a capacity of 2,200 corpsmembers, about 75% of them male, Gary is the second largest Job Corps Center.) This CAI/CMI program has separate experimental and control groups in each of three areas: basic reading, basic mathematics, and General Educational Development (GED), which is a high school equivalency program. The computerized program at Gary consists of the following hardware:

1. A minicomputer (Honeywell Level 6) which drives the CMI program and CAI lessons from the Minnesota Educational Computing Consortium and the Portland, Oregon, Public Schools and presents them on 41 cathode-ray tubes (CRTs);
2. A PLATO CAI program presented on 16 CRTs with a menu of 275 lessons specially targeted at teaching the basic skills in reading, mathematics, and GED;
3. A CAI program presented by four Apple II terminals.

Students are usually assigned to the CAI/CMI program in 3 hour blocks of time during which they have access to lessons in a variety of content areas. Specially trained peer aides are used and the teacher student ratio is approximately 20 to one.

Since input to Job Corps centers is on a weekly basis, Job Corps educational programs are, of necessity, fully individualized and self-paced, with open-entry and open-exit according to each individual's needs and progress. Upon arrival at the center corpsmembers are tested for basic achievement level in reading and mathematics using two short but accurate paper-and-pencil screening tests developed and tested by Job Corps over the past 17 years. Within a week, new corpsmembers are assigned to either an experimental or a control group in either basic skills or GED. Assignment has not been fully random throughout the history of the program, however, since schedules of corpsmembers in some vocational training areas (all corpsmembers participate in vocational programs) dictate the times they are available for educational programs. At basic skill levels most corpsmember are assigned to programs in both reading and mathematics, although they may not necessarily begin both at the same time. In order to take advantage of familiarity with the computer systems and to avoid contamination of findings, corpsmembers in one experimental model are always assigned to the other experimental model and never to the control group. Upon entry into an experimental or control model, students placed in basic skills are given the appropriate form of the Stanford Achievement Test in mathematics or reading as a pretest. Pretest scores provide a basis for measurement of gain in achievement and can be used in the statistical comparisons between experimental and control groups to adjust for any differences in entry level.

The comparisons between the control and experimental groups in this program are somewhat unique. In most experimental evaluations of CAI the control group receives conventional lecture and text instruction; thus comparisons are typically between individualized, self-paced CAI and a conventional format (Orlansky and String, 1979). In this program both the experimental and control groups receive individualized, self-paced instruction but the experimental group receives the material using a fully computerized system that offers diagnosis, prescription, instruction, practice, review, and testing. Thus, this evaluation assesses only the impact of a computer-managed and assisted delivery system.

There is one other component of the Job Corps program that makes its finding unique, but also may provide strong evidence of the value of any educational models evaluated. Corpsmembers enter into Job Corps on a voluntary basis (although some are strongly urged to attend by social service agencies, state employment services, or the courts) and are free to leave whenever they choose. Therefore, many leave Job Corps before completing the individualized program recommended for them. As stated above, the average length of stay in Job Corps is about six months, although an individual is permitted to remain in Job Corps for up to two years. Any differences in length of stay in Job Corps between participants in experimental and control models may suggest that different motivational factors exist for each program model.

It was hypothesized that there would be a greater increase in educational gain for students participating in the computer-based models than those in the control models. Educational gain for basic skills is measured by administering the appropriate form and level of the Stanford Achievement Test, converting the raw score to grade equivalent level and computing any change from the grade equivalent level obtained from the pretest. Posttests are given after 45 hours of instruction and again at 90 hours for any students still remaining in the program. Typically, 45 hours of instruction is completed in about two months time. Progress for GED level students is determined by GED practice and state test scores.

RESULTS

The results presented here are based on incomplete findings from an ongoing program. To date, information on programmatic effectiveness, in terms of educational gain, is available for students in the basic skills program in reading and, to a lesser extent, in mathematics. While large numbers of students have entered the models, smaller numbers have successfully completed them. Hundreds are still active in the programs and others have dropped out of the program before completing it or were unable to complete the program when the facilities were severely damaged by a tornado. Some of the findings have previously been reported in the Educational Improvement Effort Phase I Final Report, prepared for the U.S. Department of Labor in September 1981.

By Spring 1982 additional and more complete results will be available in all content areas. Results for reading and mathematics will be updated, and sufficient numbers of students will have completed the GED models to make meaningful analysis possible.

Reading

Since the program began in late 1980 more than 200 students have entered the CAI/CMI reading and the control model and completed or terminated from them. For both groups, the average enrollee is just over 18 years old, has had 9.7 years of schooling before entering Job Corps, and reads at a fifth to sixth grade reading level. For the CAI/CMI model, 65% of the students are black, 17% white, and 17% Hispanic; three-quarters of the students in this model are male. For the control group, 74% of the students are black, 12% white, and 13% Hispanic; 37% are male.

In order to compare the educational gains of corpsmembers in these two models it was necessary to statistically adjust for the differences between the two groups. To accomplish this an analysis of covariance model of linear multiple regression was used with race, sex, age, and entry reading level among the covariates and gain per hour of instruction as the dependent variable. With these adjustments, any differences found between the groups represent the effects of the treatment and not artifacts caused by differences in students participating in each model. Based upon an analysis of findings for the September 1981 report, which also included a second similar control group, the differences between the CAI/CMI group and the control groups were highly statistically significant ($\underline{p} < .001$), with the regression analysis explaining a rather large 44% of the variance ($\underline{R} = .66$).

To further explain and update these results some additional descriptive statistics were computed and are presented in Table 1. These results are based upon all those in the experimental and control groups for whom posttests were available in November 1981.

Table 1.
CAI/CMI Reading versus Control Reading

	CAI/CMI	Control
Number of Completers	106	38
Mean Pretest Level (Grade Equivalent)	4.8	4.8
Mean Gain (Grade Equivalent)	2.5	.9
Mean Hours of Instruction	48	45
Mean Gain per Hour	.052	.021

These results indicate that the mean gain in grade level equivalents for each hour of instruction is .052 for corpsmembers in CAI/CMI and .021 for those in the control group. Thus, educational gain is 2.5 times greater in the experimental group than the control group for each instructional hour on the average. In but a few months of instruction the CAI/CMI group increased its reading level from almost fifth grade equivalency to more than seventh grade, while the control group increased from the same starting level to less than sixth grade equivalency.

Mathematics

More than 150 students have entered the CAI/CMI and control mathematics models and completed them or terminated from them since input began nearly

1½ years ago. For both groups average age is 18.1 years, three-quarters of enrollees are male, and enrollees have completed more than nine years of school prior to entering Job Corps. For the CAI/CMI group 64% of the students are black, 17% white, and 17% Hispanic; for the control group 60% are black, 9% white, 13% Hispanic and 17% Asian. Students enrolled in the experimental group also had higher scores on the mathematics screening test than did controls, underlining the need to use caution in interpreting these early results until an analysis of covariance can be performed.

Data on the mathematics models were not sufficient for statistical analysis at the time of the September 1981 report. Descriptive statistics were computed for all students for whom posttests were available in November 1981 and are presented in Table 2.

Table 2.

CAI/CMI versus Control Mathematics

	CAI/CMI	Control
Number of Completers	120	66
Mean Pretest Level (Grade Equivalent)	5.8	5.0
Mean Gain (Grade Equivalent)	1.7	1.1
Mean Hours of Instruction	52	46
Mean Gain per Hour	.032	.024

In Mathematics, educational gain for corpsmembers in CAI/CMI is 1.33 times greater than educational gain for the control group per hour of instruction.

CONCLUSION

The initial results of the comparison between CAI/CMI models and control models in both reading and mathematics have demonstrated that educational gains are greater in CAI/CMI, as follows:

o Reading gains in CAI/CMI are significantly greater than those for a self-paced control group;

o The CAI/CMI reading group showed a gain in grade equivalent levels that is 2.5 times greater than the gain for the control group;

o The CAI/CMI mathematics group showed a gain in grade equivalent levels that is 1.33 times greater than the gain for the control group;

Other feedback on the programs suggests that one of the possible explanations for the greater educational gain for the CAI/CMI groups is increased motivation. Students report that learning on the computer is more fun and that being able to work on a computer makes them feel smart. Teachers confirm the positive motivational effects of computer-based education and indicate that students actually stand in line to use the terminals. Furthermore, the teachers themselves have become enthusiastic promotors of CAI/CMI, noting that it frees tham from paperwork and allows them to devote more time to teaching and creative classroom activities.

These findings demonstrate the success of CAI/CMI in basic and remedial education for youth. Although this demonstration was conducted in Job Corps, the results should also be applicable to other educational settings. Secondary schools, community colleges, and vocational training schools and programs may find similar successes with computer-based education, even with otherwise hard to reach students.

REFERENCES

TEAM Associates, Inc. Educational Improvement Effort: Phase I Final Report. U.S Department of Labor Contract # 99-7-998-36-17, Sept. 1981.

Orlansky, J. & J. String. Cost-effectiveness of computer-based instruction in military training. Institute for Defense Analyses, P-1375, 1979.

MICROCOMPUTER SOFTWARE
EVALUATE -- EVALUATE -- EVALUATE

Evaluation of software is one of the most important, if not the most important, process to implement when considering the use of the microcomputer in the classroom. Evaluation of software should occur when looking to purchase a microcomputer because software is hardware dependent -- that is programs designed for a specific microcomputer, like the Apple II, will only run on the Apple II and programs designed for the Atari will only run on the Atari, etc. Evaluation of software should also occur whenever the purchase of software is being considered in order to determine its probable educational effectiveness.

A number of avenues are open in which the educator can investigate the educational effectiveness of programs -- articles in computer magazines, information from organizations that evaluate software, asking other educators who have used specific programs, evaluating programs themselves, and reading field study reports. The first three avenues are relatively easy to understand. The last two, however -- evaluating programs and reading field study reports -- require further information.

In order to carry out direct evaluation of programs, a plan in the way of a form needs to be considered. This form should include the content of the program, the implementation of educational principles, and the technical aspects in the programming and the running of the program. Such questions as the following should be addressed:

1. Is the content consistent with the goals and objectives of the program?
2. Is there a way to assess the effectiveness of the program in meeting the stated objectives?
3. Is the content task analized and are the steps appropriate for the intended audience?
4. Are the pre-requisite skills, vocabulary and concepts stated and are they, in fact, the ones necessary to successfully complete the program?
5. Does the text follow accepted rules of syntax, spelling, etc.?
6. Is the product designed for the appropriate age and ability groups?
7. Is there an instructional guide and does it contain adequate information on the content and running of the program, evaluation techniques, teaching suggestions, etc.?
8. Are directions within the program clear and easy to understand?
9. Is feedback positive and appropriate?
10. Is the presentation on a personal basis?
11. Can the student run the program independently of the teacher?
12. Is branching employed to higher and lower reading and conceptual levels?
13. Does re-teaching, breaking the concept in smaller steps and changing the presentation, occur when

the student makes incorrect responses?

14. Is the format (the way information is presented) on each frame appropriate?

15. Are color, music, graphics, inverse print, and blinks used to reinforce concepts appropriately and/or for attention purposes?

16. Is the program difficult or impossible to be inadvertently interrupted by the student?

17. Is the microcomputer the best media for presenting this subject matter?

Besides direct evaluation of programs, companies and school systems can, and should, complete field studies on software to determine its effectiveness on specific populations. Field studies should include: (1) a description of the population; (2) a description of the methodology used to determine effectiveness -- questionnaire, pre- post-test, observation, testimonial, etc.; (3) a report on the results; and (4) conclusion drawn based on the results and the population on which the field study was conducted.

In designing a field study, a phone call or direct contact to the site should preceed a follow-up letter if at all possible. This sets the scene for cooperation and provides valuable information about the site and the students. Information about the sample selected should be collected regarding geographical area, grade levels, age range, how the software was presented and any other information needed to describe the site and the persons involved. The instrument used should assess whatever you want to find out -- for example a questionnaire based on a specific evaluation form. A space should also be available for the educator to add comments, test results, etc.

Evaluate -- evaluate -- evaluate! Unless software is evaluated and the educational effectiveness determined, poor software will continue to be developed, and the microcomputer with all its potential could have the fate of so many other teaching tools -- a dust collection on a shelf.

Florence M. Taber, Ed.D.
Media Design
Interpretive Education, Inc.
157 S. Kalamazoo Mall, Suite 250
Kalamazoo, MI 49007

"Study to Examine Computer Technology for Implementation into a College Reading Program"

Introduction

The purpose of this study is to examine the spectrum of computer capabilities for implementation of computer-assisted instruction (CAI) and **computer managed** instruction into the Reading Department at Norfolk State University. Norfolk State University is a historically Black urban institution located in Norfolk, Virginia. Since 1935 it has evolved from a one room unit of a private university into a fully accredited multi-purpose institution that is currently serving in excess of 7,000 students.

The university has an open-door policy and serves a student population that is characterized by a vast heterogeneity in terms of background and need. One of the most pressing needs found among the students is reading development for college achievement.

The Reading Program has the potential for excellence; however, there is a need at the university to improve the instructional management, testing and evaluation of the program for access towards more efficient diagnosing of student needs and to move quickly into the necessary programmatic changes dictated by these needs. The use of computer technology as a support system for the program is one of the possible answers for meeting these needs. Computers are now at the center of public attention and occupy a position of importance. They do so many things that people also do, but much more efficiently. They manipulate symbols that have meaning; they store information; they answer questions; they participate in the decision-making process; and they compute. In short, computers seem to possess an "Intelligence" not found in other machines. Therefore, the use of computers would greatly improve the system of instructional management, record keeping, test score evaluation, statistical analysis and reporting needed for the Reading Program.

Statement of Need

Computers are said to be the "Classroom of the future" and, in ten years or so, schools may be forced to organize classes using master teachers, aides and computer-assisted-instruction. In fact, this may lead some people to ponder the question, whether teachers will be replaced by computers. How much is needed before schools can really begin to use computer technology. Teachers would have to be trained to use computers in the classroom. Also, teachers would need to be conservant with computer software, and most programs are written in computer languages that involve training in order to understand. The need to learn about computers and how to do something with them is very important for optimum computer usage and computer applications.

There is no evidence that a computer can substitute for a good teacher, but, there is evidence that a well-developed computer supported curriculum can help less-abled or less-experienced teachers do a better job; computers can help take a heavy management load off burdened teachers and administrators; computers can reduce costs and increase efficiency as they have done in industry. When educators can accept the use of computers and develop maximum usage from them, the winners will be students, teachers, schools, and society as a whole.

Presently, entering freshmen

students are placed into the Reading Program depending upon scores received on a battery of tests given at the beginning of a semester. The objectives, expected outcomes, and program activities provide continuous need for assessment in determining progress and planning for individualized instruction. Often, test scores are hand scored and recorded entailing time consuming tasks for the laboratory directors and instructors. If the task of scoring tests is relegated to the Testing Bureau they, too, would have to record the scores via hand. Also, the student demographic data for information and placement needs to be more centralized for individualizing instruction within the Reading Program. The implementation of Computer Technology could assist greatly in this effort by improving the management for the activities of the program and enhancing the academic delivery system of the University. The computer can process instructional diagnostic prescriptive techniques and recycle models which will accelerate student achievement and eliminate much of the hand scoring and paper type record keeping. These activities can serve as a supplement to the regular university testing program.

The university is now using a Hewlett Packard 3000 to generate data for a large segment of administrative functions. An additional computer for academic programs is to be obtained by the university, which will allow for terminal access for academic computing, for this project. The use of computer terminals in the departmental office would allow the appropriate personnel to feed information into the terminals for maximum articulation and provide an easy access to the equipment.

Procedures

The implementation of computer technology into the Reading Program can provide for effectively individualizing instruction, improving instructional management and coordination and facilitate the reporting and recording of test score information, thereby, providing immediate feedback for the formative and summative evaluations of the program.

Computer Assisted Instruction. The benefits of a CAI program can assist in making learning an enjoyable experience for the student and the teacher. Generally, CAI provides individualized, self-paced instruction that can significalntly upgrade student performances, be highly motivating, allow an increase in student/teacher ratio without the student's performance being adversely affected, help teachers better diagnose areas of student weakness, ensure the application of proven teaching methods to all students at all times, and provide immediate feedback to the student.

Management and Coordination. As demands for accountability and improved educational effectiveness increase, the need for effective management and coordination becomes more pronounced. This activity will implement the mechanism for management and coordination within the Reading Program through the use of computers (hardware) and the existing data base management (DBM) program (software) to produce and keep track of an organized collection of test scores that will be used for instructional management.

Testing. Perfunctory progress reports due to poor test interpretation hinder the effectiveness of reading progress for the recycling of instructional activities. Solutions to the problem of test score interpretations and test score reporting can be addressed in measuring related test score data and recording test scores and printing various statistics about the test scores. Programs can be made to add students, delete students, add tests, or modify test scores such as changing a score after a make-up test is given, etc.

Evaluation

The evaluation activities can provide clear information about how effective such programs are and what factors contribute to their outcomes. Good evaluative information can improve instructional practices and management. Evaluation in this study refers to both the collecting and reporting of test scores as well as the assessing of activities upon program improvement. Techniques for assessing program outcomes from a number of perspectives including appropriate attitudinal and social measures and varied cognitive areas of the reading program can be developed.

The evaluations will consist of (1) Student evaluation of laboratories, (2) Student evaluation of teaching, (3) Evaluation of laboratories by administrators and faculty campus-wide and by administrators and faculty within the program, and (4) Self-evaluation of faculty in comparison to a locally developed profile of the "ideal" reading instructor (personal characteristics, skills, etc.). The profile will be developed by the faculty and staff of the Reading Department.

The computer can program techniques in data arrangement, file organization and access, modular and structure programming and select methods for insuring effective program design, testing, documentation, and evaluation. The data collection techniques can provide a variety of evaluative insights into program processes.

Conclusion

The Reading Department is thoroughly committed to implementing computer technology into the program structure at Norfolk State University. Presently, the Department Head is enrolled in an introductory course in computer science and some faculty members have completed a three month on-campus workshop presented by the Computer Center. The School of Education, which includes the Reading Department, is planning programs for teaching pre-service teachers about computers and educational technology. The Reading Department is planning to take an active role in all three endeavors.

References

Berquist, William H., Ronald A. Gould and Elinor Miller Greenberg. Designing Undergraduate Education. San Francisco: 1981

Doerr, C., Microcomputers and the Three R's. Rochelle Park, New Jersey: Hayden Book Company, Inc., 1979

Dusseldvip, R.V., and D.W. Spuck, Microcomputers: Their Selection and Application in Education. Michigan: Association for Educational Data Systems, 1979.

Mason, G. E. and J. S. Blanchard. Computer Applications in Reading. Newark, Delaware: International Reading Corporation, 1981.

Sanders, Donald H., Computers in Society. New York: McGraw Hill, Inc. 1981.

Spencer, Donald P., A Guide to Teaching About Computers in Secondary Schools. Ormond Beach, Florida: Abacus Computer Corporation, 1973.

- Carmelita Williams

INSTRUCTIONAL MANAGEMENT: CIMS-IMPACT CASE STUDY

Charles Adams and Kathryn Cameron

I. The CIMS System

Four years ago the educators of Oneida County (upstate New York) were requested to investigate the feasibility of a regional management system approach to mathematics education. The theory that mathematics education need not vary from district to district was a novel if not a rather unnerving concept. Loss of local control, the "watering down of an already fine district program", were among the concerns voiced by educators on the regional approach. Teachers were used to making curricular and instructional decisions based upon their perceptions of student needs rather than documented data and a large pool of pedegogical expertise.

A recent revision of the New York State Education Department Bureau of Mathematics syllabus served to point out that many elementary teachers were unaware of the existence of the State guidelines and an even greater number chose to ignore them in favor of the curriculum offered by a selected textbook. Thus, the commercial publisher (13 different vendors) had unwittingly become the final authority on mathematics education for local educators.

The CIMS (Comprehensive Instructional Management System) Project offered an alternative to the above mentioned approach. The concept of a county wide objective based curriculum based on the syllabus provided by the New York State Bureau of Mathematics required that the process of development be as carefully considered as the eventual desired product. Each Chief School Officer (14 original districts) selected a district representative (generally a teacher with strong faculty rapport and good mathematics background) to serve on the initial project development committee. Although the committee later expanded to accomodate the magnitude of the effort, most of these original representatives still provide the fundamental strength and expertise to the current effort. The initial school year (1978-79) was spent developing terminal, grade level mastery, and specific objectives in elementary (K-6) mathematics. The specific objectives were logically grouped (modules) and organized into a scope and sequence chart. Appropriate activities were written for the objectives and finally, test items as well as mastery criteria were determined. The CIMS project was ready for district implementation in September, 1979. Over the past three years, revisions have taken place and enhancements provided. The addition of prerequisite activities and the keying of manipulative materials to the objectives served to further build local ownership as well as to aid teachers in utilizing the project. Alternate tests were constructed to parallel existing module tests for pre-instructional and makeup situations. The development with the greatest implications for the CIMS project was the introduction of Survey Tests. Key objectives were selected at each grade level and students were tested prior to instruction in September. As the students move through the school year, they are again tested following instruction by the module tests. In June, the Survey test is again administered to check student retention of objective mastery.

This process of curriculum development and instructional methodology is similar to the mastery learning concept advocated by Bloom (1976). The curriculum used by all teachers is a well specified set of learning outcomes (instructional objectives), teaching-learning activities designed specifically to teach a given objective and criteria referenced measurement to determine student skill mastery. The use of an information system to provide periodic feedback to students, teachers, parents and adminstrators was an essential aspect built into the system. Without this information, informed decisions concerning individual pupil learning, instructional strategies and curriculum appropriateness could not have been made With such a large amount of data needing to be analyzed very frequently the need for computer support to make the system operational was soon evident Thus, the marriage with Project IMPACT was made.

II. THE IMPACT SYSTEM

IMPACT is a computer based student achievement information system that maintains instructional assessment data on pupils longitudinally. The system has four major components, a Master Curriculum File (MCF), a Criterion Referenced Test Scoring and Analysis subsystem (CRT), a Norm Referenced Test Scoring and Analysis subsystem (NRT), and a Student Achievement Historical File (SAHF). The system is designed as an integrated data base (TOTAL data base manager) written COBOL to run on large IBM main frame computers (370 or 4300 series). It provides for both batch and certain on-line processing applications.

Three curricular elements, instructional objectives, CRT measurement items and teaching-learning prescriptions (CIMS math curriculum is an example) are stored in the MCF. The CRT component scores the modular mastery tests given to students and stores, objective by objective, resultant mastery data for each student. This component provides a number of periodic and summary reports for teachers, administrators and parents useful for instructional decision making.

The NRT component scores and analyzes pupil responses to almost all popular standardized norm referenced tests (e.g. IOWA Test of Basic Skills, California Achievement test, etc.). The fourth component, SAHF, stores individual pupil results of standardized tests. This is a cumulative history file covering as many school years as desired which allows for longitudinal progress assessment of individuals and groups of pupils.

The CIMS project utilizes three of the four components of IMPACT with the greatest use being made of the MCF for curricular assessment, the CRT for monitoring individual pupil progress and instructional planning by teachers, and the NRT for external summative project evaluation.

At the end of each instructional module the teacher administers a criterion referenced mastery test to which the students respond on optical scanning answer sheets. These are then sent to the computer for scanning and scoring. The output reports of the CRT module provide teachers at each test occasion: 1) an individual student report indicating the student's mastery status on each objective tested, 2) a class diagnostic report that groups all students in a class section by objective(s) they have failed to master, and 3) an item analysis of the test form used. At other intervals and upon specific request, local school personnel also receive a group summary report that indicates how all students in a course or teacher section are performing on all course ojectives at both pre and post instructional points in time. They can also obtain a year-to-date summary report for each pupil on their mastery status by individual course objective and including which teaching-learning prescriptions have been assigned. If desired, as in the CIMS project, student achievement data can be aggregated and reported across a number of districts which use a common curriculum.

III. RESULTS

Results of the CIMS/IMPACT effort are significant. Student achievement of a coordinated sequential flow of mathematical concepts has become the goal of educators instead of a certain number of pages to be covered in a text. The teacher is charged with meeting student needs rather than the student being required to meet teacher expectations. This change in focus from content coverage to the student mastery of skills represents a major shift in thinking that has had positive results.

Although standardized tests do not represent the best diagnostic tool of a criterian reference based system, the results of the administration of the California Achievement Test to all CIMS participating districts have been positive. A brief summary of the county wide results for all students, K-6, are provided in the table below:

Table 1

	Bottom 3 Stanines	Middle 3 Stanines	Upper 3 Stanines
79-80	5%	50%	45%
80-81	2 - 4%	50%	46%-48%

The conversion of the New York State Pupil Evaluation Program from norm reference to criterian reference measure in the Spring of 1983 will provide additional external evaluation data on the CIMS project.

Perhaps most significant is the ever expanding look that the data provided by the management system allows into the status of mathematics education. Without pre-instruction data, many children are given instruction on concepts already understood prior to instruction. It has been found this occurs particularly at the primary level. The information provided by the September administration of the CIMS Survey Test is useful in helping the teacher determine teaching priorities.

It has been further determined that topics in which teachers are apparently insecure are simply not being taught in many instances. (Probability and Statistics as well as metric education are in greatest need of inservice.) All key objectives with less than 80% retention mastery (determined by the June administration of the Survey Test) are listed and made available to appropriate adminstrators in each district. All objective information is available on a classroom, building, district or project wide basis. It is now possible to investigate mathematics education by topic and/or grade level as well as provide complete objective mastery documentation on all CIMS participating students.

Another assessment project was undertaken by the Project IMPACT development team and they found similar results for a group of teachers using a reading program. Again, two successive years of student achievement data were collect and analyzed. This analysis used the criterion referenced score data, as opposed to the norm refenced assessment conducted by the CIMS project. Student achievement gains, in terms of master of objectives assigned and taught, were evident in each class section analyzed. There was, of course, much variance in the percent of objectives mastered from one student to another, some mastering all of the course content while others had demonstrated mastery on only a small percent of the objectives.

Comparing the results of two years of implementation of the IMPACT system showed an increase in the average percent of course objectives mastered by students in the second year over the first year at most grade levels. It would appear that as school personnel gain greater familiarity with, and understanding of, the information provided by the system their use of it has a positive effect upon student learning. Schultz (1979), in her study of teachers' adaption to innovations, found that some teachers reported that it took four years to get acclimated to using periodic testing and a computer system. And, Howland (1981) indicates, teachers who had specific training and a high level of understanding in the IMPACT System found the information much more useful for instructional decisioning.

Finally, whether the information provided by the computer based achievement information system did directly contribute to improved student achievement through improved instructional decisioning by educators using the system or not, it does provide an information base that can be used for instructional decisioning. The fact that the system does provide for achievement record keeping and periodic reporting on individual student progress as well as summative data on program effectiveness means that the information is now readily accessible to those who choose to use it for those purposes. This information support capability has been the primary development objective of the project and its developers and pilot test users believe that objective has been met. Its complete utilization for instructional decisioning rests more with those who are to use it on a regular basis than the nature of the software system itself. This implies that greater attention in the future must be given to staff training and implementation strategies than computer system design and development. Hopefully, in the future these aspects will become the focus of evaluation and research.

References

1. Bloom, Benjamin S. - Human Characteristics and Learning. New York: McGraw-Hill, 1976.

2. Howland, Ruth. Teacher Reaction to the Use of a Diagnostic-Prescriptive Approach to Classroom Instruction Employing a Data Automated Student Achievement Support System, Paper Presented at the American Education Research Assoc. Los Angeles, April, 1979.

3. Schultz, Dorothy Grant. Teachers Adapt to Innovation. ERIC Document #ED185495, 1979.

DEVELOPMENT OF A STATEWIDE SYSTEMS ARCHITECTURE: EXPERIENCES FOR THE STATE OF MINNESOTA

SITUATION

What can a State do when its 13.4 million dollar investment in large scale hardware systems, and approximately 3.5 million dollar investment in application software falls short of increasing demands for computer service from educational users? A dilemma indeed, but there may be a solution in sight--an approach to progressing from highly centralized processing with older batch applications to more decentralized processing with flexible accommodation of varied user needs exploiting newer technology available in microcomputer systems. In this paper we'll explore a systems architecture for integrating hardware, communications, and applications in a cohesive and comprehensive manner.

The development of educational computing in Minnesota began in 1967. In 1970 a strategy for provision of computer equipment and information systems statewide was developed. From 1973 through 1979, the State developed management information systems for administrative applications. These applications support a district user base in Minnesota which has a broad profile: (a) 3% are districts of greater than 10,000 students; (b) 18% are districts of between 2,000 and 10,000 students; and (c) 79% are districts with less than 2,000 students.

Regionally centralized computing facilities are the major strategy for this delivery system. The regional delivery system includes equipment, staff, and a governing board operating with requirements and software developed centrally. Such software exists for finance, payroll/personnel and student systems. Burroughs B6000 series equipment is the large scale processor at the service center. Users may be connected to the central processor by a CRT, or remote job entry (RJE) equipment, or may be serviced via data delivery and pick up services.

Current demands for local access and control have always resulted in acquisition of large scale processors for centralized computing to support on-line entry. In addition to the large scale equipment base, application systems had been designed to operate in a batch mode. Rework of the systems to provide for on-line update and retrieval would require an extensive commitment of scarce personnel resource. However, demands for service from users are pushing the State into some form of distributed processing. These demands resulted in two problems. First, not only is system redesign expensive, but it will not solve the problem of a sophisticated user base: the desire for control over data processing. Second, Minnesota school districts have had extensive experience with microcomputers in instructional application, and desire to exercise that "in-house" level of control with administrative computing.

There are several disadvantages to the current strategies. The most significant disadvantage is high resource consumption at the host processor to support on-line access during the day. The use of this on-line approach, in which terminals are controlled as slaves within a host controlled, polled network, results in heavy daytime demand on the host processor. CRT's and remote job entry (RJE) devices perform little or no preprocessing and are driven by host loaded formats. The devices have no local storage facilities and generate substantial overhead and high communication costs. Although these terminals provide for rapid host capture of information from outlying locations, responsibility for error detection, editing, and data validation resides at the host, increasing resource utilization and causing processing delays. In the case of districts serviced by delivery service, turnaround times for each of the activities of data collection, error correction and reporting must be considered.

THE DILEMMA

Current budget constraints in the State of Minnesota demand that any future strategy for the delivery of data processing service for administrative computing must be substantially cost effective. Minnesota's "dilemma," then, is how to create a state-of-the-art network which will support varied and complex user needs, take advantage of technological change, and allow for a smooth transition to a new software environment.

USER NEEDS

User information needs have driven the solution to this systems architecture. These needs for the elementary, secondary, and vocational information system have been divided into three general levels: simple; intermediate; and complex.

Simple information requirements typically need little automation, except where standards of reporting have defined a more rigorous collection method, such as Minnesota Uniform Financial and Reporting Standards (UFARS) have done for district financial accounting. However, for the purposes of categorization on the basis of information needs, those districts desiring only to comply with the minimum UFARS requirements are identified as having simple needs.

Intermediate information requirements necessitate some automation, but this level does not require inquiry into current sets of data for the management of district operations.

Complex information requirements are highly dependent on automated support for district decision making. Complex levels of information require selection, summarization, and display capabilities.

The determination of simple, intermediate, or complex information needs is to be made by each individual school district based on its unique information requirements, which may not be contingent on size alone. For instance, some districts may have information needs in one area (payroll) and intermediate or complex needs in another (student).

The same division of user needs is used in this discussion of application systems design options. The options are best divided into three levels of information requirements:

- simple;
- intermediate; and
- complex.

For finance systems, those with simple information should have two options available. These options should be:

- microcomputer for data collection/edit; and
- microcomputer Financial System (MICRO-FIN).

For the intermediate level information requirements, there are two options:

- microcomputer for data collection/edit; and
- small satellite for data collection/edit.

The microcomputer data collection/edit routines, described under the simple information requirement, are an option. As with the simple information requirement, detailed transactions would be transmitted to the region for batch processing.

The option of a small-scale satellite would provide for data collection and edit similar to that provided with the microcomputer data collection and edit. This edited data would then be communicated to the host processor. In a future phase, an additional capability could be added to support file transfer from the host to the small satellite where file creation/maintenance/inquiry and report generation could be supported. The small-scale satellite is the lowest level at which local intelligence could support limited file manipulation for output reports from subsets of the host data base. (Note that the MICRO-FIN would be a stand-alone system and would not interact with the regional host except for annual reporting requirements which would be gathered at the regional center and transmitted to the State Board of Education.)

At the complex information requirement level, there are three options:

- microcomputer for data collection/edit;
- medium-scale computers with a "down-scale"* version of the finance system resident; and
- medium-scale computers with "down-scale" versions of Finance, Payroll/Personnel and Student Systems.*

The microcomputer with dial-up to the medium-scale systems is the same approach described previously for data collection and edit for the simple and intermediate levels. This system is tied in via dial-up lines to the host. Medium-scale satellites would be configured with the front-end edits for the Finance Payroll/Personnel and Student Systems. Limited local processing and reporting would be supported.

At the next stage, the medium-scale computer would provide a subset, or "down-scale" version, of the Finance System. Front-end edits from the Payroll/Personnel and Student Systems would be supported and these systems would remain at the host.

At some future date it might prove feasible to "down-scale" versions of the Finance, Payroll/Personnel and Student Systems and operate them on a stand-alone basis at the medium-scale satellite. Microcomputer data entry to the medium-scale satellite would continue to be supported.

HARDWARE

The architecture is based on a mix of microcomputers, small-scale satellites, medium-scale processors and large host processors which can fit coherently to support district information requirements.

---

* Creating "down-scale" versions of the administrative systems to operate on medium-scale host processor will not be without cost. The architecture states a migration path: districts must decide that this path to local control is worth the cost.

Microcomputers

Microcomputers will be used to: (a) provide front-end edit of district batch data; and (b) stand alone support for a finance system compatible with state financial reporting standards.

Small-Scale Satellites

Small-scale Satellites will be used to support intermediate level information requirements, including

- front-end edit, validation, and data collection; and
- file creation/maintenance/inquiry.

This size processor is capable of locally supporting data files, including subsets of the data base at the regional host computer. Using this capability, the front-end data collection and edit routine can be expanded to include data validation. An example of data validation would be a routine of checking financial transactions to ensure the transactions are in conformance with the chart of accounts.

With the small-scale satellite the district should have the capacity to report using subsets of district data which have been downloaded from the host processor. These files could then be used for district inquiry.

Medium-Scale Hardware Applications

Those districts that have complex information requirements may have the need or desire to operate independently on a medium-scale computer. Such an approach would have to be implemented in stages which are financially achievable by the district requesting those capabilities. The local computer might initially have front-end edit and validation routines resident on the machine. At the next stage, stand-alone, down-scale versions of the Finance system could be used to support district operations, with the regional host computer continuing to be used for the Payroll/Personnel and Student Systems. During the final stage, down-scale versions of the Payroll/Personnel and Student System might be used in this stand-alone operation at the district.

At all three levels the host is off-loaded of data entry and correction routines. Simple financial information requirements can be met with little or no host processing load by means of a stand-alone MICRO-FIN application. Data entry and correction remains at the user site. The security and simplicity of the batch processing design is retained at the host processor. The MICRO-FIN would probably be the only application which is off-loaded from large- or medium-scale hosts, although microcomputer payroll/personnel and student systems exist in the State currently and might become attractive to other districts.

These approaches do have disadvantages and are not without cost. The first is that a microcomputer version compatible with the state's financial requirements must be completed. The second is that intelligent data entry front-ends for the three applications systems must be built. The final disadvantage is that down-size versions of the applications must be developed. The first two disadvantages have been surmounted, the third requires substantial work and would be warranted only if a district deemed its requirement complex and was willing to bear the financial burden to implement the new application.

BENEFITS OF THE SYSTEMS ARCHITECTURE

The benefits that will accrue to the State and to district users from the adoption of this Elementary, Secondary and Vocational (ESVC) architecture are:

- improved use of installed equipment and available resources;
- improved overall quality and flexibility of services provided to the school districts;
- more direct school district control over resource allocation and service alternatives; and
- minimal incremental investments in hardware, software, communications and other resources.

Some cost reductions for the present hardware and communication architecture have also been projected. While similar savings have not been projected for software development, greatly improved flexibility should be available to the district user for a comparable level of cost.

The advantage of this hybrid approach to other organizations in a similar situation are that: (a) considerable investment in software and hardware can be extended in usefullness into the future; (b) extensive benefits are gained in user friendly data entry; and (c) future hardware acquisitions can be based on incremental cost versus large-scale lease/purchase commitments.

- Theodore Bahn

THE IMPACT OF MICROCOMPUTING ON SECONDARY SCHOOL ADMINISTRATORS

Larry Bowers, Principal
Downers Grove North High School
Downers Grove, Illinois

The secondary school administrator (Principal, Assistant Principal, or Assistant Superintendent for Instruction) faces several significant challenges during the '80's because of the advent of the microcomputer. One of these challenges is the psychological impact of the micro on teaching staff. A confidence needs to be built in staff who do not have computer backgrounds. These staff members need reassurance that they can learn about educational applications of microcomputers. One of the techniques that has been found to be successful in imparting this confidence to teaching staff will be described in this paper. Our high school developed as a building goal the theme of developing computer awareness in the teaching staff. In an effort to accomplish the above goal the Staff Development Committee of administrators and teachers developed teacher inservice, institute days and staff development programs around this goal. One of the outcomes of the staff inservice committee was the development of a staff development program entitled 'The Introduction to the Microcomputer, Part I, and The Introduction to the Microcomputer, Part II.' This program was designed by this committee and the Principal of the high school. Microcomputer I, Staff Development Program, became a ten-hour graduate course. It was offered from 3:30 p.m. until 6:00 p.m. on four consecutive Tuesday evenings during the month of September. When the class was announced to the faculty the subscription to the course was overwhelming. Since the school was in possession of only four microcomputers, it was decided to limit the class enrollment to sixteen teachers. This would allow a 4 to 1 ratio of teachers to microcomputers.

The outline of the Microcomputer Staff Development Program is as follows: The purpose of the program is to inservice instructional staff on microcomputers with a hands-on type of course which would enable staff to plan/develop/utilize existing software in the instruction of their courses. The workshop title would be 'Microcomputer Workshop No. I - Introduction to Microcomputers.' This workshop would present a thorough introduction to the microcomputer, its peripherals, and beginning basic programming. This workshop would be designed for teachers with little or no computer experience. Participants who have come through this workshop will possess the following skills:

- Come to understand the use and application of the microcomputer and its peripherals.
- Learn to safely connect and disconnect peripherals.
- Learn to store and run programs from disk or tape.
- Learn to list a program on the printer.
- Be able to print messages on the monitor screen.
- Be able to input responses to a program from the keyboard.

The four class sessions of the workshop can best be described by presenting the outline of each session. These outlines are as follows:

I. SESSION I - INTRODUCTION

A. Introduction

169

B. Historical Perspectives
C. Present the Equipment
D. This Is How It Works
E. Laboratory Session

II. SESSION II - EDUCATIONAL APPLICATIONS
   A. The Microcomputer As A Calculator
   B. The Microcomputer For Drill and Practice
   C. The Microcomputer For Simulation
   D. The Microcomputer As A Teaching Aid - i.e. Generating Tests
   E. The Microcomputer For Data Analysis

III. SESSION III - PROGRAMMING - WHAT TO LOOK FOR IN SOFTWARE - PROGRAMMING TECHNIQUES - THE REAL TRUTH ABOUT COMPUTER PROGRAMMING
   A. It Is Very Simple To Do
   B. To Become An Expert Takes Just About Much Time As It Takes To Become An 'Expert Pilot' (draw analogy)
   C. A Program Is A Tremendously Efficient Tool (do sample print program)
   D. A Program Can Become More Complicated (Do Manhattan Island Project Program)
   E. Use of Filmstrip (Show two 20-minute portions of a filmstrip on the B.A.S.I.C. computer programming language) (stop filmstrip at suggested break points for the purpose of clarifying, explanation, and for questions)

IV. SESSION IV - SPECIAL FEATURES
   A. Printer
   B. Color Graphics
   C. Sound
   D. Graphics Table
   E. Card Reader

As referenced earlier in this paper, the subscription to the above Introduction to Microcomputer course was tremendous in our school. As you can tell by the outline of the course (one 2½ hour session per week for 4 weeks--total of 10 hours) it would take one month to complete one section. During the initial sign-up stage for the Staff Development workshops over 50 staff from a staff of 120 teachers signed up. The first workshop was held during the month of September, 1981. After the first workshop was over more people signed up and there was a demand for a Microcomputer II workshop. At the time of the writing of this paper it is March of 1982. At this point in time over 75 staff members have gone through Microcomputing I workshops and we are now in Session II of Microcomputing II. I feel the secret to the involvement of our staff in microcomputing is based upon several factors. In my opinion they are as follows:

1. Microcomputing is a timely topic in society at the present time. The public does not quite realize the difference between a main frame, a mini, and a microcomputer. There is even confusion between Intelevision, Atari, and microcomputers. This gross confusion causes a tremendous natural interest in the field.
2. The topic of microcomputers is generally thought to be comprehensible and understandable. The staff that might be suffering from fatigue, burnout, or just

170

overwork find something with an excitement about it sort of self-renewing.

3. Teachers are always curious to find new techniques for presenting their information.
4. Many staff feel that they become programmers and ease the workload in their classroom.

Whatever the reason for becoming involved in microcomputing, it has become one of the most exciting staff development activities that our school has ever experienced. I would recommend it without hesitation.

After 50 to 60 staff members completed the Microcomputing I Staff Development Program, a great deal of interest in using the microcomputer in departments developed. Teachers' confidence level had risen to the point where they were comfortable in integrating the microcomputer into their class sessions. This integration was most frequently accomplished by utilizing purchased software. The following list indicates software that is used by teachers in their classes on a frequent basis:

Business Education
AMIDON (Consumer Education)
Business Education Packet
VISICALC Apple II

Computer
Advanced Applesoft
Apple files
Applesoft special features
DEMO
Design Vols. I & II
Introduction to Applesoft
Pascal Vol. I
Programmer's Aid Vol. I
Apple Dissemination Disk #1,2,3

Developmental Education
The poet

English
Reading Comprehension: What's Different? 32K
Three Mile Island

Foreign Language
Apfeldeutsch - German Routines
The French Hangman
The Spanish Hangman

Home Economics
Eat Smart - Nutrition Program

IMC
Apple Seed Computer Literacy Package(Discovery, Applesoft, Tutorial)
Classroom Management DK-Class A Disk Library
Elementary Vol. 2, Version 2.4
Social Studies Vol. 2, Version 4.2
Word Root System

Industrial Arts
Apple Gradebook
Apple World (Graphics pkg.)
Automotive Diagnosis
Electrical Engineering, I & II
Flight simulator, Version 2.0
Locksmith (Appledisk Copy Program)
The New Step by Step; an Interactive Course in BASIC
Zoom grafix

Mathematics
Computer Graphing Experiments, Vols. 1-3
Genis I Program for Computer Mathematics Vol. 1, Version 3.0
Shape Tables Vol. 1
Teacher Utilities Vol. 1, Version 3.1

Music
Arnold (Level I) Program/ Applesoft BASIC 48K, Version 1.1
Harmonious Dictator Program/ Applesoft BASIC 48K, Version 2.0
Melodious Dictator Program/ Applesoft BASIC 48K, Version 2.4
Music Composer with DAC Board/ Applesoft BASIC 48K, Version 24
Rhythmic Dictator Program

Science
Motion (College Physics Series Vol. 3)
Science Vol. 2, Version 3.0

Science Vol. 2, Version 3.1
Science Vol. 3, Version 4.3

Administrative Uses Of The Microcomputer:

The microcomputer's potential as an administrative tool is tremendous. School districts across the nation are using online attendance systems on the Apple, TRS80, and other micros. Other administrative applications of the microcomputer are:

1. To control the operation of classroom bells--the microcomputer may be used to isolate the ringing of cafeteria and resource center bells on a lunch period schedule. This technique saves the interruption of classes throughout the building by the ringing of lunch period bells. This is accomplished through utilizing a simple time schedule program that sends a TTL (transistor type logic) impulse through a tiny relay which trips a larger 12 volt relay which eventually activates the ringing of selected cafeteria and resource center bells.
2. The microcomputer Management of keys--The entire inventory of keys to classrooms, desks and files may be added to the microcomputer. This particular application allows a school administrator to have a complete listing of each key for each staff member. It also provides a current inventory of the number of keys that are available for each lock. This is a tremendous management tool for a school administrator. The security conscious school administrator can effectively control access of the building by utilizing this program.
3. Student locker management through the use of the microcomputer--Programs have been developed for the microcomputer that allow the student locker number and combinations to be put on file in the microcomputer. The students who are assigned to a particular student locker can also be entered into this file. This gives the administrator in charge of student locker assignments the capability of having at their fingertips the number of each locker, the name of students in a particular locker, the current combination of a locker and a maintenance exception report for student lockers.

Administrative applications with the microcomputer open up a new management alternative to school administrators who previously were unable to make effective management decisions because of the lack of information organized in the appropriate manner.

COMPUTER SUPPORTED STUDENT MANAGEMENT FOR A LARGE CLASS

Thomas A. Boyle
Professor of Engineering
Division of Interdisciplinary Engineering Studies
Purdue University
West Lafayette, Indiana 47907

In a recent paper[1], Dr. Richard W. Davis of the Bell System Center for Technical Education presented a hierarchy of computer applications in education. In this, he identified five approaches to such use:

1. Computer maintains records, does registration, scheduling, etc.
2. Computer supports testing
3. Exercise materials provided by computer
4. Computer provides or facilitates access to course or reference materials
5. Computer provides instruction

Davis went on to note his need for "a working environment on the computer which is the equivalent for me of what the machine shop has been to the tool and die maker. I need powerful tools, precise tools, big tools, small tools; tools I can use to make the instructional programs I conceive of as an educator."

To whatever extent you are familiar with machine shop practice, my hope is that you will regard my proposal as of one of Dr. Davis' needed tools. My proposal relates most specifically to the second item in the hierarchy, computer supported testing, however, its influnce spreads over several others and there are some side effects which I hope you will deem desirable. But before introducing my topic, let's consider some details of its environment. Such considerations will enable you to judge whether the new model 'turret lathe' I propose is suitable for duty in your 'machine shop.'

The class situations serving as reference here typically involve 400 to 800 students. For at least part of their treatment, the students are placed in sections, each with enrollment ranging from 35 to 80. Collection and assessment of student work and test scoring are done centrally. Punched cards are used for input of data pertaining to student homework. A card with student identification punched in prescribed format is submitted with each job. The evaluator notes the score on the job itself, for return to the student, and on the accompanying card for subsequent punching and entry. After each increment of student work, computer prepared reports are produced for each section. These reports include cumulative data so students can confirm the system record of their efforts to date. Test response data are entered either by keypunch or via opscan to tape. The test scoring program produces a summary report for each section and an individual report for each student. Most of these procedures would not currently be regarded as noteworthy, however, some features of information gathering at test time, and the feed back of the information, may merit your consideration.

The special capability of the computer application discussed following is the preservation of information that is necessarily lost when all information from a test is encoded into a single grade or score. Preservation of the information enables identification of student difficulties and preparation of atypically detailed reports, both to individual students and to section instructors. Of course, in order to be preserved, the information must first be obtained. This is done through the combined capabilities of programmed testing and computer scoring. Each of these complements the other. Without computer scoring, the test programming would be impractical. Without programmed testing, the capability of the computer is wasted.

The unusual feature to be considered, the programmed testing, differs from customary practice in two ways. First, when taking a test, each respondent encounters in suquence one of several categories of test items. The sequence is practically the same for all respondents. It makes little difference wether items are answered correctly or not. Next, each item encountered must be answered. When students encounter items which they do not understand, they must guess answers in order to go on. One cannot strive for a high test score by working only familiar items. By forcing students to attempt items in each of several categories, the programmed test makes students reveal relative strength and weakness.

The computer enables scoring this type of test in a manner which would otherwise be practically impossible. Each student's score is accumulated in parts or subtest scores, each corresponding to one of the categories of test items. The scoring program adds a point to the appropriate subtest score for each correct response. Occasional mistakes are not penalized, but whenever an error is detected, a flag is set in the related subtest. If the next item encountered in the same subtest is missed, a point is subtracted from the subtest score. By thus penalizing errors in sequence, the scoring program yields expected scores near zero for students adopting a strategy of pure guessing. A student who consistently misses related items will get a negative score indicating test performance worse than guessing. When testing any typical student group in this way, some students usually obtain negative scores in one or another subtest.

For those who insist on student scores suitable for ranking, the scoring program simply adds the subtest scores. An additional step in scoring is the division of each subtest score by the number of related items attempted. The resulting ratio scores have been shown to stabilize after a student works a modest number of items, and thereafter remain insensitive to variations in test administration time.

The profile of subtest ratio scores, together with the number of items attempted, provides the information needed for the computer to prepare a message to the student. Evaluation of the score profile can be either criterion- or norm-based, with generation of a message advising students how they score with respect to the selected reference. In other situations, for example following testing related to remedial work, the computer-prepared message can be based entirely on an individuals score profile, offering congratulation, or at least encouragement for high scores and noting need for extra effort as appropriate in other places.

The information presented in Table I is representative of a part of a report as it would be provided to a section instructor. The scores shown as examples are from a 30-minute arithmetic test administered to post-secondary technical college students. Six subtest scores are shown for each student. The sum of the six scores appears in the Total column. A ratio score appears in decimal-fraction form just below each integer-number score. These ratio scores indicate the proportion of correct responses. For example Sam earned

TABLE I

Example Programmed Test Score Profiles

	Subtests						
Name NF	1	2	3	4	5	6	Total
Sam	3	-4	2	1	-4	1	-1
37	.43	-.57	.33	.17	-.80	.17	-.03
Bev	5	-2	-2	-5	1	-4	-7
44	.83	-.22	-.29	-.63	.14	-.57	-.16
Tim	8	7	6	-1	2	2	24
48	1.00	.78	1.00	-.13	.29	.20	.50
Pat	7	10	7	-7	8	8	33
48	1.00	1.00	1.00	-.88	1.00	1.00	.69

Ratio scores appear as decimal fractions below each subtest score. NF = total number of items attempted.

a score of 3 in the first subtest. Inasmuch as his ratio score there was 0.43, he evidently answered 3 out of seven items correctly. Interpretation of negative scores is not quite so straightforward, but these always indicate consistent errors. Sam's score of -0.57 in subtest two indicates he missed six out of seven items attempted. His score of -0.80 reveals that he missed all five items he attempted in subtest five.

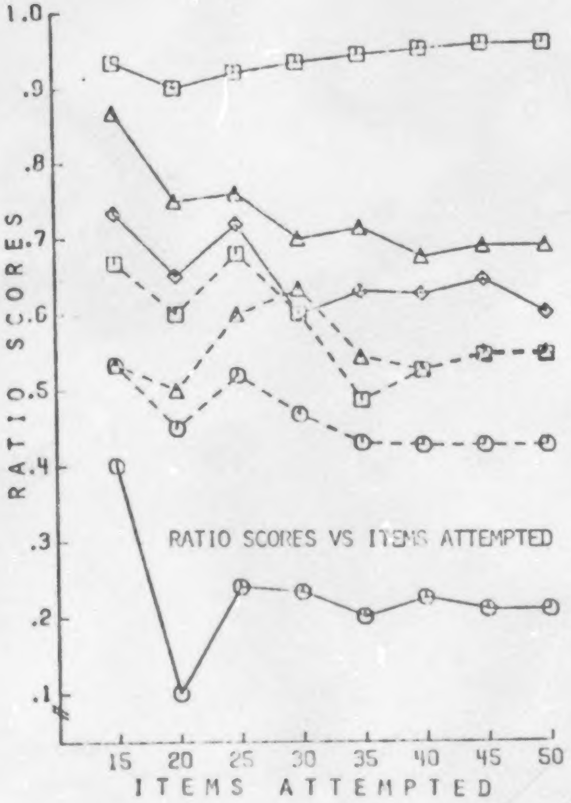
The principal point to be gained from all this is that the combination of programmed test and computer scoring conveys considerably more information than otherwise would be practical. Jonathan Warren[2] has noted the unfortunate loss of information which results when all information regarding student achievement is encoded into a single score or grade. The same is illustrated in Table I. If one were to consider only total scores (i.e., if all six subtest scores were encoded into one) then Bev, the second student listed is seen to earn a substantially lower score than Sam does. But attention to the subtest scores shows Bev has answered five of six items correctly in the first subtest. Indeed if subtest criterion were 0.8, then Bev has demonstrated adequate grasp of the material represented in that subtest, although she does worse than guessing in all but one other.

The score profile shown in Table I for Tim indicates acceptable test performance in the first three subtests, concurrent with decidedly lower scores in the other three. If the six subtest scores relate to a sequence of instructional modules, then Tim has presented evidence showing him to be midway in the sequence. Finally Pat's scores present an unusual profile. In terms of her total score one would judge her to be only marginally adequate. But Pat's subtest score profile shows she answered all items correct except those in subtest four. Here she makes mistakes on all eight items she attempts. The combination of programmed test and computer data processing enables feedback of such detailed information, as shown in Table I, to instructor of supervisor. Futhermore, it is a simple matter to utilize a students score profile as the basis for a computer-prepared report, effectively advising the student of high and low points and recommending appropriate action.

Again pursuant to Davis' hierarchy, a frequent extention of computer support for testing is into the business of providing exercise materials. Programmed test materials adapt readily here. Students with only rudimentary knowledge of computers can be provided system access adequate for scoring their efforts on a paper and pencil test version. Alternatively sections of programmed tests can be accessed via terminal. In either instance a moderate size section of programmed test material can yield a large number of alternative forms for exercise, simply by designating different start points in the programmed array. Some of the items used for exercise can be included in the test used for summative evaluation. Here, for example, the scoring program could distinguish between the score on items previously available for exercise and those being encountered for the first time. In many instances the prior access to a set of test items has been observed to make little difference in the distribution of scores obtained.

Certain collateral advantages accrue to the use of computer processed programmed tests. First the combination enables before-after testing with the same instrument. Although no supporting evidence is offered, the retest effect appears to be less than would be observed on a comparable conventional test. Students taking the same programmed test a second or third time frequently obtain scores lower then they did the first time. Second, the combination of computer and programmed test does an efficient job of getting information with regard to the amount of administration time. Experimental work reported[3] showed that 77 percent of the information obtained from a 100-minute test was obtained in the first 25 minutes.

The score characteristic with respect to administration time manifests itself in another way. The accompanying figure shows the trend of ratio scores for several students who work rapidly when taking a programmed test. Here the ratio scores are observed to remain practically unchanged during the last half of the administration period. This suggests that ratio scores for programmed tests are practically immune to minor variations in adminstration time. This characteristic is desirable when using the same test for summative evaluation in several different sections. A conventional test, in which students can work on items they choose, will likely yield a different characteristic. The conventional scores obtained in a section given extended administration time would be atypically high. But, as the students run out of familiar items and necessarily attempt more difficult ones, then ratio scores would probably decrease as administration time is extended.



A further side effect resulting from programmed testing is its inhibiting effect on the transfer of information between students during testing. Even when students begin at the same point in a programmed test they seldom work the same sequence of items. The different sequences of items can be made practically equivalent, yet the responses one student writes on his answer sheet will be almost entirely different from those of his neighbor.

Exchange of helpful information between students in one section and those in another is similarly inhibited. Students having already been tested are usually able to convey only generalities about the test they have had. If they do convey information regarding a specific test item to a student in a section to be tested later, then the probability that the student will encounter the specific item discussed is roughly one out of three. When the same programmed test is administered to multiple sections there appears no evidence of advantage accruing to the last sections tested. In one such administration to 14 sections of a chemistry course no significant between - section differences were observed. The distribution of scores obtained by students tested on Friday was practically identical with that obtained on Monday. Evidently this capability of computer supported testing mitigates the need for alternative test forms or for scheduling multiple section together for test purposes.

References

1. Davis, Richard W., "Where The New Technologies Are Taking Us in Training," College Industry Education Conference, San Diego, 1982.
2. Warren, Jonathan R., "Grades as an Information System," American Society for Engineering Education, Event 2530, June 1978.
3. Boyle, Thomas A., and Gary L. Wright, Computer Assisted Evaluation of Student Achievement," Engineering Education, Vol. 68, No. 3, December 1977.

Developing a Faculty Computer Literacy Project in a Multi-campus Community College

Barbara Guthrie-Morse, Ph.D.
Valencia Community College

OBJECTIVES:

Three major objectives are explicit in this report. First, with a very broad brushstroke, I want to paint a picture for you of the need for a community college faculty computer literacy project. In painting that picture, I will reflect on national, state and local trends and developments that directly influence that need. Secondly, within the context of the Florida Staff and Program Development (SPD) framework, I want to outline for you a faculty computer literacy project methodology for a multi-campus community college environment. The SPD program provides an excellent and appropriate vehicle for stimulalating faculty interest in reviewing instructional courseware and in developing minimal design/development skills. My third objective is to share with you some of the initial experiences of this infant project and some long-term projected continuation plans. Clearly, one of the major goals of the first project cycle will be to stimulate faculty interest to the point that additional SPD and other project proposals will be developed independently.

PERSPECTIVES:

In the first phase of the College Board's National "Project Equality," computer competency is referenced in both the mathematical and the study skills competencies. In both lists, the computer is viewed as a learning tool for problem solving and as a resource in learning how to learn.

At the state level, the "understanding of computer technology concepts" is one of 21 groups identified as "college-level computation skills associated with successful student performance through the baccalaureate level. "Computer-technology understanding will be measured through the identification of (1) "characteristics of tasks which computers usually perform well, (2) the human functions necessary to utilize computers, and (3) possible abuses of computer use."

At the local level, the Orange County (Florida) Schools utilize more than 200 personal microcomputers for classroom instruction and management. In this area, high school graduates will be more sophisticated than the two-year college graduate.

That these national, state, and local initiatives follow closely on the advent of the low-cost micro or personal computer is not coincidental. Now computer-assisted and/or computer-managed instruction is realizable in the college's open instructional labs and classrooms. The computer can extend the instructor by performing the repetitive drill-and-practice routines, by providing immediate feedback, and through individualized instruction.

Put simply, I believe that computers in education are revolutionary because they make possible great teaching in a system dedicated to mass education. But they make this possible by supporting person-to-person educational influence, not be replacing it.*

*Thomas Dwyer, "Some Thoughts on Computers and Greatness in Teaching," Topics in Instructional Computing, Vol. 1, ACM Special Interest Group on Computer Uses in Education, 1975.

SETTING:

Valencia Community College is a multi-campus, multi-cultural institution that serves some 12,000 credit and non-credit students each fall and winter session. Valencia employs 170 full-time faculty members and from four to five-hundred adjunct faculty members in the instructional process.

Valencia offers an A.S. degree program in Data Processing. The required instructional computing capacity is provided by Valencia's main-frame and a Wang minicomputer with four terminals. An instructional data processing lab is available on both the East and West campuses. Currently, the college's main-frame computing capacity is being administratively reviewed. Until recently, Valencia had no microprocessing capacity. One East Campus faculty member utilizes his own personal micro for physics instructional experimentation.

Valencia faculty are interested in alternative instructional methodologies as evidenced by faculty participation in open instructional laboratories and in the development of competency-based instructional methodologies. On both campuses reading, writing, math, and accounting open labs serve numerous students each session.

Specifically, the Valencia faculty has demonstrated an interest in exploring computer-assisted instruction and computer-managed instruction. Faculty have attended conferences, workshops, have taken course work, and have submitted small SPD proposals geared toward developing computer expertise.

Over the last two years, Valencia has hired two key instructional campus deans who agree that a critical institutional priority must be established for faculty computer literacy development. These leaders represent almost all of the full-time faculty and have taken an active leadership role in stimulating interest and discussion of a common college project. The first funded SPD college-wide project is the result of that initiative.

THE SPD PROJECT:

The currently funded SPD project totaled some $11,000.00 and included four major activities spanning the winter and summer 1982 terms at Valencia. The project activities include the following:

1. Faculty release time to review courseware, to review microcomputers, and to make recommendations for micro-processor purchase.
2. Courseware acquisition to be supported by a minimal budget allocation for each campus.
3. A faculty computer literacy course to be designed, developed, and implemented in early summer.
4. Site travel and conference attendance to be promoted through SPD-allocated funds.

The project will be managed by a faculty member who will be advised by a college-wide steering committee which will include the two instructional deans and faculty representing various disciplines.

RESULTS/CONCLUSIONS:

The project proposal has been written and has been funded for the first cycle. Now some 3 months into the project, some seemingly obvious caveats have emerged. Before considering such a project in your institution you would be wise to consider the following:

- The range of knowledge about and interest in computer tech-nology is apparently unbounded.

- The most knowledgeable faculty are all ready the busiest faculty.

- The best technology, the best systems require trust and credibility for productive use.

- The best technology requires sound, comprehensive, accessible educational programs to garner acceptance.

- The acceptance of technology must be established. It is not inherent.

- Users must view computers as a useful tool *not* see themselves as extensions of the machine.

To date, steering committees have met, a faculty survey has been developed, the first computer literacy course is being designed, and a second funding proposal is being written. Some 20-30 Valencia faculty attended the February instructional computing conference, two micro-computers have been secured and are operational, and some software has been acquired. Further, the College has utilized the considerable faculty expertise in developing a substantial, multi-year federal project proposal. Finally, *substantial* professional reading regarding computers and possible external funding sources is being shared college-wide.

BIBLIOGRAPHY

Fasciona, Josephine. "Can We Afford to Say Good Morning? Mixing People and Computers in the Workplace." *VITAL SPEECHES of the Day*, February 15, 1982, pp. 277-280.

Wiegner, Kathleen. "Tomorrow Has Arrived,"*Forbes*, February 15, 1982, pp. 111-118.

Schwartz, Helen J. "Monsters and Mentors: Computer Applications for Humanistic Education," *College English*, February, 1982, pp. 141-150.

Cognitive Style and a First Course in Computer Science: A Success Story

Johnette Hassell
Computer Science Department
Tulane University
New Orleans, LA 70118

Abstract: Theorists in software design have argued for some time that programs should be designed in a top down, or hierarchical, fashion. Yet two significant problems pose difficulties in using this approach for first computer science courses: there has been no tool, available to beginners, which supports and encourages top down designing; in addition, hierarchical methods of problem solving are foreign to students with certain cognitive styles (of problem solving). This paper reports on a tool being used in a first course in computer science, which is a service course to all departments in the university, and its success in assisting students with diverse cognitive styles. Also reported are the results of an experimental evaluation of the use of this tool in introductory computer science courses.

BACKGROUND

Cognitive style is a measure of how one approaches learning and problem solving. An understanding of cognitive style began in the late 1940's. Numerous experiments have added to the understanding of its nature. One common way of measuring cognitive style is via the Embedded Figures Test, in which a subject is presented with a complex figure and is asked to find the simple figure embedded in the complex one.

When presented with several such pairs of figures, some persons are consistently able to find the simple figure faster than others. These faster solvers are, in some sense, able to filter out the distraction of the additional lines in the surrounding field of view.

Other experiments require subjects to align objects (lighted rods or their bodies) with true vertical when their perception is confused by tilted frames of reference.

These studies show that those most distracted by the excess lines of the complex figure in the Embedded Figures Test are also most influenced by the frame of reference in other situations; i.e., those slower at finding the hidden, simple figures also adjust their bodies or the rods more off vertical than others with faster time in the Embedded Figures Test. Thus, at one extreme, subjects are strongly influenced by the clues and structure of their surrounding fields and are called "field dependent." At the other extreme, subjects seem to function by some internal standard and ignore extraneous clues in their surroundings. These persons are called "field independent." It is of note that the field question may be visual, physical, intellectual, social or auditory and yet the distinctions between field independent and field dependent remain.

The measure of field independence/field dependence seems to be a basic style used in a variety of perceptual and intellectual situations. Thus it is called "cognitive style." Cognitive style is, then, a measure of a collection of preferred ways of viewing ones surrounding, thinking, learning, and problem solving. It investigates questions of form rather than content of cognitive activity.

180

The problem solving strategies of relatively field independent and field dependent persons vary markedly. A relatively field independent person makes use of any organization of the information of a problem. If the material is poorly organized or has no organization at all, the field independent person modifies the stucture or imposes a new one on the problem. A relatively field dependent person, in contrast, will continue to work with the given organization.

In one study (Fleming, 1965) students were asked to remember a series of words, which happened to be hierarchally related. When the words were presented in reverse order, from the bottom of the hierarchy, field independent students were significatly more successful in remembering the words than field dependent students; ie., they formed hypotheses about the structures and used this structure as a memory aid. When the words were presented from the top of the hierarchy, there was no difference in the performance of students with different cognitive styles; the field dependent students were able to use the obvious field structure to remember the words.

Since we are, as a culture and as scientists, biased toward more field independent cognitive styles, it is important to note that there is no correlation between cognitive style and intelligence. For example, in one study, both field independent and field dependent subjects took the same number of tries to master the material, though they varied in rate at which they learned (Nebukoph and Dreyer). In addition, numerous studies have shown that cognitive style is not a predictor of college grade point average (Anderson, Pohl).

While cognitive style is self-consistent, it is only an indication of an individual's preferred mode of problem solving. Alternate behaviors can be adopted (Whitkin, et al) particularly if it is appropriate to the problem itself. In the following sections, a method of software design which supports the continuum of cognitive style is presented. It is shown to provide a way for more field dependent students to develop better problem solving techniques.

A critical phase in software development is design. At this stage, it is determined how the computer will be used to solve the problem. This includes the development of algorithms and data structures to solve the problem.

There are two reasons that software design is of such central interest. Poorly designed software is responsible for incredible costs in terms of time, money, computer resources, safety and personal frustration. Secondly, software designing is a creative, problem solving activity. This makes it particularly difficult to teach. In some sense, a teacher must rely on the student's own inherent problem solving ability and approaches.

A design tool is a method of assisting in the design of software. It may or may not be automated. It consists of a notation and rules for applying it. The result of the design process is a design document or simply a design.

Tulane uses a highly structured design tool called Structure Charts in its first year courses. After using structure charts for several semesters, the author became aware that more field dependent students, who traditionally do poorly in first programming courses (Cheney), were doing very well in their programming courses. This led to the hypothesis that the programming of more field dependent students would be improved by the organization that structure charts imposed on programming.

An ideal experimental design would have included a matched control group taught using more traditional, ad hoc, programming procedures. However, the Tulane committment to the structured approach is so strong that a control group on campus was deemed unacceptable. Thus the experiment includes two groups of students, those in the first semester of their sophomore or senior year of study respectively.

All the sophomores had two semesters of instruction using structure charts as their design tool. Some of them had had "progamming" in high school before their entrance into college, but most had not. The seniors had been taught programming by older, ad hoc procedures. Most had not used structure charts extensively. The exceptions are those who have worked as a lab assistant/grader for the first year courses.

All participants were given the individual Embedded Figures Test. In this test, field independence is indicated by smaller average times. In addition, all participants also took a two-hour test designed to measure their understanding of programming and programs.

It is difficult to use program writing in a test situation as an evaluation of a student's ability. First, the test situation is not like real life in that students usually are able to use system tools, such as editors and compilers, to avoid various errors. These are not available in the test situation.

Writing programs in the test environment handicaps those who work methodically and carefully (but slowly) in designing software, but who produce better work because of fewer errors and better design. Testing also tends to obscure the difference between designing a program and coding it because the emphasis is on producing a program to be graded. Grading then becomes difficult because it is often impossible to distinguish between a syntax error, a logic error, and/or logic errors caused by a syntax error.

Thus, the experimental measurement did not include program writing. Students were asked to perform two kinds of tasks. The first part asked students to memorize a program. They were told that they had to be able to produce a functionally equivalent program. Variable names did not have to match. There were three such memory programs; two had to be reproduced in a different language, and the third in the same language (FORTRAN IV, FORTAN77 and Pascal are the languages used.)

The second part asked students to find errors in programs. They were given the choice of FORTRAN77 or Pascal. Five programs had to be debugged. Students were allowed to correct errors or merely indicate where they were.

RESULTS

A total of 28 sophomores and 19 seniors took both the Embedded Figrues Test and the programming evaluation. In all the statistics that follow, it is important to note that the seniors have two more years of experience in programming in both academics and summer jobs.

The sophomores have averages of 21 and 10, out of 29 and 19 respectively, on the memory and debugging tests. The seniors have averages of 25 and 12, respectively. It is meaningless to ask if these differences are statistically significant because of the difference in the backgrounds of the two groups. The closeness of the scores, particularly on the debugging exam, does indicate that these exams may be a useful measurement of the understanding of programs and should be explored further.

Of interest to the author are the correlations between average time on the Embedded Figures Test and scores on the exam. Since a faster time on the Embedded Figures Test is expected to correspond to a higher score on the test, negative correlations can be expected. Those correlations for sophomores are -0.063 and -0.067 for the memory and debugging exams, respectively. For seniors, the correlation coefficients are -0.49 and -0.48.

There were some reservations concerning the debugging exam since the two groups contained students who had served as graders. These roles required them to debug student designs and code. When the students with grading experience are excluded from the data, the correlation between the time on the Embedded Figures Test and the two-test-total score is -.03. For the seniors, the correlation rises to -.53.

Thus, for sophomres there is no relationship between cognitive style and success on the programming measurements. However, there is a moderate one between the cognitive style of seniors and this measurement, despite their increased experience and education. While this latter correlation is not statisticaly significant, it looms large in light of the difference between the populations.

We conclude, then, that the use of the highly structured techniques of Structure Charts provides the more field dependent student with a structure to impose on the problem. In light of the work on remembering words presented hierarchically, it seems that presenting a form which induces a hierarchy on the problem further enhances the more field dependent student's ability to understand programs and programming.

NOTES

This paper represents an abridged version of the report of this project. For the full paper and references, contact the author directly.

ACKNOWLEDGEMENT

The results of this project are heavily dependent upon the contributions of Laurie G. Foley, Andrea Lawrence, and Judith Walters, who helped to administer Embedded Figures Tests, design computer science tests and compile statistics.

The ABC'S of Farming
or
Agriculture By Computer

by

Thomas R. Hintz
Computer Network Manager
University of Florida
Institute of Food and
Agricultural Sciences

The selection of a topic for this paper puzzled me for a while. How can agriculture relate to the theme of this conference, "Computers in the World of Education"? After a time it became apparent that education was more than just formal training in a classroom environment. Providing useful information to a person in need is another form of education.

The title, although appropriate for an educational group, should also convey a less obvious meaning. Specifically, it means that the use of computers for agricultural related functions has just begun to materialize. We have only started to "plow new ground" (scratch the surface) for utilizing the potential benefits of this complex technology and are just beginning to develop and learn our new ABC's.

Why computers and agriculture? You might think that this is an unusual combination of industries. However, the bringing together of these dissimilar technologies is imperative. Why? Because a grower of food can no longer be considered just a farmer, but is instead a producer. This is more than just a change in terminology, like a garbage man being referred to as a sanitary engineer. This change relates to a grower's changing function.

At one time farming was simple; punch holes in the ground with a pointed stick, drop in seed, cover with soil, and let mother nature do her thing. This lifestyle is adequate when only concerned with feeding just yourself and your family. But with the continuing shift from an agricultural society to an urban society, two things are happening. There is a decline in tillable land available for food production, and an increasing burden on fewer producers to satisfy our demand for food. As more and more people become dependent upon these crops it becomes increasingly important to boost productivity. Consequently, usable land must be utilized more efficiently.

We have already experienced the "agricultural revolution". This term was used by Alvin Toffler in his book entitled, The Third Wave (1). The "agricultural revolution", the first wave, allowed us to keep pace with our demand for food. Toffler's second wave, the "industrial revolution", improved mechanization and provided additional capabilities to further improve production efficiency. His third wave, the present "technological revolution" (which includes computers), has not been fully utilized throughout agriculture, but will become increasingly important.

As part of this third wave, we have entered a new era, the era of information. Information that, because of its rapidly expanding volume, requires new ways to record and communicate what we have learned. This information is both complex and dynamic, and should be included as part of the decision making process. As a result of this information explosion, the complexities of producing and marketing agricultural commodities have become very demanding. Therefore, producers are starting to use computers in ever increasing numbers to provide some relief.

What has stimulated this activity in agricultural computing? Basically, two things. First of all, the infusion of computers into our daily lives is a result of the declining cost of computer power. Who could have guessed that the processing power of the first computers, that cost several million dollars, consumed large amounts of energy and occupied an entire room can now be purchased at an affordable price, run on batteries and carried in your pocket. The easily recognizable names of Apple and Radio Shack desktop microcomputers, to name a few, are now being found in many growers homes and offices.

The second reason for the flurry of interest in computers, and the most important, was brought about by our need to know. Computers are now able to provide information that was previously unavailable or difficult to obtain. It is now possible to view temperature data transmitted from satellites to determine the potential for freezing weather in an orchard, simulate crop and pest development to determine optimum harvest and control strategies, and to quickly search large data files for current regulations for control of pests. All the examples given above deal with very dynamic data. This volatile data is important in making value judgments about the everyday operation of a farm. One mistake can be very costly, both to the grower and the consumer.

To help in this decision making process, many Land Grant Universities have been actively developing agricultural application programs. In a survey conducted by Dr. Robert Strain (2) of the University of Florida, approximately 400 individual programs were available in 1980. In a recent update to this survey, this number has more than doubled to a total of over 1000 programs having been developed. Many other programs are available from private companies. This is only the beginning.

Several years ago, the Institute of Food and Agricultural Science (IFAS) anticipated the added demands that would be placed on it for providing agricultural information. Being a state funded agency, IFAS has the responsibility of addressing the needs of Florida agriculture. But before it could effectively serve its clientele, it needed to devise a method for providing up-to-date information throughout its own statewide distribution of offices.

At the University of Florida (U of F) campus, IFAS is composed of 21 separate departments, the College of Veterinary Medicine and the School of Forest Resources and Conservation. Although based at the U of F, IFAS is also responsible for the support of the research, teaching and extension activities at 9 Agricultural Reasearch and Education Centers (AREC), 14 Agricultural Research Centers (ARC), and Extension offices in 67 counties.

Because of the widely dispersed locations of IFAS personnel, it has become increasingly difficult to meet the rapidly growing demands for information on an institute-wide basis. An new type of communication system was needed. A system that could provide rapid collection, analysis and dissemination of information.

During the last five years, IFAS has been formulating a mechanism for meeting the current and future needs for agricultural information. After an extensive evaluation of programmatic areas it was determined that to increase state-wide computer availability to IFAS faculty and as a prerequisite to the implementation of certain state-wide programs, a network of minicomputers is required. Working with several other state agencies to ensure continued compatibility with existing computer equipment without the duplication of existing capabilities, a technical network specification was developed and approved.

At the present time, most of the computing by IFAS is done at the Regional Data Center (RDC) located at the U of F. This RDC supports several IBM mainframe computers. Lesser amounts of computing are also done on several small minicomputers located in a few departments in addition to a handful of microcomputers scattered among several county offices, all of which are dedicated to specific research tasks.

It has been stated that "the most cost effective use of modern technology is to place the computer power in the hands of the user, where and when, it's needed." This has been shown to be the trend throughout the computer industry. In a recent Business Week article entitled Moving Away From Mainframes (3), it was shown that because of "...the far lower cost of these small computers and the convenience in using them for dedicated applications, customers have begun to buy them in large numbers."

The article goes on to state that "these new corporate strategies do not mean that the market for mainframes.... is dead. The growth of distributed systems will create a new role for mainframes as electronic librarians that will manage huge, centralized data banks." Our network design will take advantage of what others have found to be the most cost effective approach to computing.

Our design approach is to develop a hierarchical system composed of several different size computers. At the top of this hierarchy will be the RDC. It will provide the greatest amount of computing power throughout the system and will continue to be used for large data base programs, data analysis, and other applications suited to the unique capabilities of this mainframe computer.

Within IFAS, and located on campus, will be a large minicomputer. This minicomputer must be large for several reasons. First, it must support several high speed links to the RDC for file transfer and program submission capabilities. Secondly, it must support the largest concentration of staff for program development and general use. Thirdly, it must support immediate access to numerous shared data bases. And last of all, it must support communications with on-campus departmental terminals and minicomputers, twenty of the off-campus offices in addition to three other medium size, remote minicomputers that provide links to the other (approximately 60) off-campus offices.

Off-campus there will be three medium size minicomputers. They will be located at the Quincy, Lake Alfred and Belle Glade ARECs. Each machine will act as a focal point for the western, central and southern part of the state. Their functions are to provide a moderate level of computing power to their assigned segment of the state in addition to providing direct access to the larger computers on campus, when required.

The lowest level machines, the microcomputers, will reside in individual offices and interconnect to one of the larger minicomputers. The complete interconnection of this array of computers will allow for simple, local processing and access to the larger machines for more complex problem solving and data analysis.

This network of minicomputers would serve as the major communications link for providing immediate access to data bases and computing capabilities that are currently inaccessible to off-campus users. In addition, it would provide a common mechanism for the collection and distribution of information to all IFAS staff whether they are located on campus or in a remote county office.

This strategy is not unique. Other Land Grant Universities have been heading in this direction for several years. Some have already succeeded. Indiana, for example, has an operational network that interconnects all of its over 100 agricultural offices throughout the state. Several other states have begun to follow their lead and already have partially operational systems.

It is believed that the creation of this type of a computing environment will allow and encourage the development of new and innovative applications in agriculture. With the dynamic changes in Florida's agriculture, these innovative applications will be a key factor to the long term growth of agriculture within the state. Only after we have integrated the ABCs and farming will it be possible for us to continue making substantial advances toward maximum productivity, our XYZs.

INFORMATION CITED

(1) Toffler, Alvin. 1980. The Third Wave. Bantam Books, New York, NY. 537 pp.

(2) Strain, J. Robert. 1980. Survey of Availability of Micro and Mini Computer Software. Food and Resource Economics Department, IFAS, University of Florida. Staff paper 165, 39 pp.

(3) Moving Away From Mainframes. Business Week, February 15, 1981, pp 78-80.

THE APPLICATION OF DISTRIBUTED PROCESSING AND NETWORKING AT BROWARD COUNTY SCHOOLS TO MANAGE STUDENT RECORDS AND PERFORM INSTRUCITONAL MANAGEMENT

FRED H. KAUFFMAN, DIRECTOR
Management Information Systems
The School Board of Broward County, Florida
1300 West Las Olas Boulevard
Fort Lauderdale, Florida 33312

ABSTRACT

The School Board of Broward County is currently proceeding with the installation of the BRAINS network (Broward Remote Administrative and Instructional Network for Schools). This network when completed will consist of 170 large minicomputers installed in all schools, centers, and major administrative offices. These computers will be networked into central site computers in order to transmit summary data and provide access to central site data bases.

This network will streamline the data collection process and improve the timeliness, accuracy and availability of data. Initially, the distribution of student records and instructional management applications out to the schools will be implemented. Subsequently, the network will be used to collect and transmit financial and personnel information.

This presentation will focus upon the justification, technical approach, and implementation steps for a network of this size and complexity.

BACKGROUND

The School Board of Broward County (SBBC) is currently the nation's tenth largest school district. It serves a population body of 127,000 students through 158 school locations. These locations break down as follows: 22 high schools, 28 middle, 97 elementary, 2 vocational and 9 exceptional centers. School based management has been in place for several years. Briefly, this means that decision making responsibility and authority as it relates to course offerings, staffing, and budgeting resides with the Principal. The curriculum is competency based and is called the Pupil Progression Plan (PPP). This means that a student must pass 70% of skills at grade level in order to be promoted to the next grade. Monitoring of these basic skills is reported by the teachers to be quite time consuming and tedious.

The central site computer complex currently consists of two 4341-1 computers operating in shared DASD mode under DOS/VSE Rel 2. There is approximately 3.6 billion bytes of direct access storage. The central site scanning is performed on NCS 7018 and 7015 scanners. Approximately 100 terminals are installed throughout the schools, county offices, and MIS. Of these, 26 terminal/printer combinations are installed in the high schools and vocational centers. These terminals are utilized for the DISCOVER career guidance system as well as for inquiry against the student data base. The secondary schools use keypunches to create their student record transactions. These transactions are then sent via inter-office mail to the computer center where they update the student data base. The elementary schools submit their student record input via OMR scan sheets. These sheets are also submitted to the computer center where they are scanned and then used to update the student data base.

In 1978 a criterion referenced instructional management system called TRACER from CTB/McGraw-Hill was implemented to support the PPP by monitoring basic skills in reading, writing, and math. The input to TRACER consists of tests which are scanned and scored centrally. The output has proven quite useful to the teachers. However, the best turnaround that can be provided is 7 working days from when the teacher gives the test until the results are returned.

NEEDS ASSESSMENT

The problems of supporting a large decentralized school district such as Broward are significant particularly when the data is submitted in batches via inter-office mail.

Some of the problems are

- turnaround time is measured in days
- error correction process is cumbersome
- system outage affects all users
- predefined production schedules must be consistent for all schools
- pinpointing data entry problems is difficult
- ad hoc reports must be scheduled

Three years ago the MIS department analyzed problems affecting its level of support such as these and performed a detailed needs assessment. The result was a set of goals defining a new long term direction. Briefly, the major goal was to define and develop an architecture which would ensure that student and instructional data be timely, accurate, and available in the appropriate format for the appropriate user audience.

STRATEGIC DIRECTION

To support this goal a five year long range plan was developed. The most significant aspect of the plan was the emphasis to be placed upon distributed processing as opposed to the large mainframe central site approach. This plan was presented to the School Board in March 1978 and approved. This approval marked the official beginning of distributed processing at SBBC.

The strategy chosen was to install a minicomputer in each school attached to a scanner and with sufficient terminals and printers to enable school based personnel to enter, update, and retrieve data related to students. This network of minicomputers would be electronically linked to the central computer site so that summary data could be transmitted for management and external reporting. This implementation strategy is termed distributed processing.

TECHNICAL APPROACH

The selected approach is for each of the 158 school locations to contain a minicomputer attached to a scanner. The configurations will range from 4 terminals and 2 printers in the high schools to 2 terminals and a printer in the elementary schools. The central site will contain 2 host minicomputers. These minicomputers will be used for file to file transfer of information with the remote network. Each day the schools will utilize their minicomputers to enter student data, maintain a local student data base, print out reports and perform instructional management. At night the host minicomputers will autocall each site and retrieve the day's student record transactions. This data will be passed to the 4341 mainframe via tape and used to update the central student data base. The edit/audit reports will then be passed to the minicomputers via tape. The host minicomputers will subsequently autocall the remote sites and transmit their reports. Then in the morning the schools will print out their reports. The file to file transmission process will also be used to transmit object code program updates and data files. The Houghton Mifflin instructional management system called ANSWER will be installed on each minicomputer. The schools will be able to score tests and monitor the status of all program objectives for each student for the major curriculum areas.

The high schools, middle schools, and vocational centers will also be linked to the 4341-1 mainframe via a CICS bisync multi-drop line. This will enable these schools to utilize DISCOVER and to perform interactive inquiries against central data bases.

The functions this network will be able to perform are summarized below:

- interactively create and retrieve student data resident at minicomputers
- interactively retrieve student data resident at central site computer
- scan and score tests
- print out reports locally both scheduled and ad hoc
- perform word processing
- transmit summary data to central site computer
- submit computer runs requiring heavy resources to central site computer for execution (i.e., computer scheduling)
- interactively use DISCOVER (career guidance system)
- enter/validate data to be submitted to central site computer

PROGRESS TO DATE

Year	Accomplishment
Fall 1978	Installed TRACER instructional management system to support Broward's Pupil Progression Plan in reading, math and language arts
Fall 1979	Developed prototype system consisting of a Northern Telecom 445 minicomputer attached to a scanner to demonstrate distributed student records and instructional management functions.
Fall 1979	Pilot tested ANSWER on a Basic/4 minicomputer. Test involved 8000 students in 16 selected elementary and middle schools. Curriculum area was mathematics.
Spring 1980	Installed student data base to include student identification, attendance, schedule, grades, and exceptionality information.
Spring 1981	Installed capability for terminals in all high schools to access information from the student data base.
Spring 1981	Completed feasibility study and user requirements document for distributed processing network to handle all student records and instructional management functions.
Fall 1981	Pilot tested student registration via a network of 10 Northern Telecom 405 minicomputers electronically connected to a Northern Telecom 445 minicomputer.
Fall 1981	Defined technical approach and documented technical requirements for the network.
Fall 1981	Issued request for proposal for the complete proposed network for minicomputers.
Spring 1982	Moved the 10 Northern Telecom 405 minicomputers to 10 selected elementary schools and expanded the applications to include the capability to maintain a local student base, perform attendance, and print class lists, rosters, labels, and ad hoc reports.

PHASED IMPLEMENTATION PLAN

I. Initial Pilot Tests

Pilot test student registration via minicomputers in 10 selected schools. Additionally, pilot ANSWER for 8000 students in 16 selected elementary and middle schools.

II. Issue RFP

Develop request for proposal for 167 minicomputers to comprise entire network.

III. Expanded Pilot Tests

Expand pilot test at 10 selected schools to also include attendance reporting, local student data reporting, and the printing of labels. Expand ANSWER pilot to handle 30,000 students in 6 curriculum areas.

IV. Award Bid

Evaluate and award bid to vendor for network hardware.

V Installation of Equipment

Install minicomputers at all 170 locations over a 7 month period. These minicomputers will initially perform student registration, attendance, reporting, label printing, and word processing. Secondary schools will additionally have access to central site student base and DISCOVER.

VI Secondary School Data Collection

Secondary schools will be provided with the capability to enter schedule changes, grade changes, graduation progress, master schedule, and course dictionary data. This will enable all keypunches and sorters to be phased out.

VII Install ANSWER at Selected Sites

ANSWER will be installed at the 16 sites used in the ANSWER pilot

VIII Expand ANSWER

ANSWER will be expanded to all school locations.

IX Distribute all Student Data

All student data for a given school will reside and be maintained at that school. Only summary data and data required by several locations will be maintained centrally.

PROJECT MANAGEMENT

In order to increase the probability of successful implementation, steps were taken to choose an approach to project organization that would fix responsibility, ensure continuing user involvement, and expedite high level decision making. To meet these objectives an MIS project team with defined functional responsibilities, an Executive Steering Committee, and a user resource group were assembled.

Major milestones were developed and detailed task/responsibilities were defined.

The MIS project team is functionally organized as follows:

Requirements
Processing Requirements
Data Base/Data Dictionary
Security/Audit/Control
Scanner Requirements

Procurement
Procurement (including scanner)
Staffing Needs (MIS)
ANSWER Contract
Ongoing School Costs
Warehouse Inventory Procedure for Supplies

Network
Network Installation
Supplies Coordination
Vendor Coordination
Transmissions
Hardware/Line Problems
Site Preparation

Systems Development
Standards
Design
Development
User Manuals
Backup/Recovery Capabilities

User Support
Inservice
Staffing Needs (Schools)
User Coordination/Help Desk
Site Startup
Backup/Recovery Coordination

Technical Support
Minicomputer Software
Remote Job Entry
Communications Software
Special Routines

In addition, once the bid is awarded the vendor's systems engineer will also become a member of the team.

The Executive Steering Committee meets every two weeks and is comprised of the Superintendent, key members of his immediate staff, Chair Principals and the MIS Director and the project leader of the project. The objective of this committee is to prioritize subsystems for implementation, review progress, assign resources, and resolve any policy issues.

The user resource group consists of all district directors and selected principals. This group forms a pool of resources upon which the members of the project team can draw as they work on their assigned activities.

BENEFITS

Decisions can be based upon accurate, available, and timely data.

Decisions of a broad scope will be based upon reliable information.

Costly impediments to smooth information flow will be dimished, resulting in greater productivity.

Teacher and system paperwork burden will be reduced.

Curriculum can be more effectively monitored and altered to meet student needs (effective delivery techniques can be disseminated.)

School-based managers will have the tools necessary to manage a school.

Reinforcing Reading Skills Using the Microcomputer and Specialized Software Designed by Elementary and Middle School Teachers

Diana Lagotic, PH.D.
Lorraine Losch
Lynn Domenech

The success of our microcomputer program in reading began at the county level when leaders in the school system became keenly aware of the challenge offered by computers as a learning tool of the future. Initially, inservice training was offered for county staff and school board members, who then demonstrated an eagerness to support teachers wishing to become involved. Next, decisions were made regarding the placement of hardware in two pilot schools.

A key factor considered when choosing the pilot schools was the flexibility of the staff, especially that of the principals and the teachers involved. Each possessed special qualities: an openness to innovative ideas, the adaptability necessary for implementing a new program, an enthusiasm which lead them beyond the call of duty, a desire to learn something new and a challenge to teach it. Prior knowledge of micros was not essential, and as a matter of fact, none of the selected principals and teachers were computer literate.

WHERE TO START?

Decisions on hardware selection were based on future availability of educational software as indicated by the manufacturers. Training of the pilot teachers which was accomplished with relative ease, involved mastering the operation of hardware and evaluating the existing software. The difficult task was to find software which met the immediate needs of the target student population.

It was found that software was limited to disciplines outside the area of reading and that readability was above the level of remedial reading students.

SECURING SUITABLE SOFTWARE

The solution to securing suitable software involved two possible options: (1) to modify existing programs or (2) to develop original software using teacher input. Both options were implemented successfully by using parent volunteers and consultants for programming assistance. While reading teachers emphasized necessary concepts and content, programmers concentrated on program design. In developing software, teachers recognized a need to correlate programs to the county basal series. Some software allowed practice and drill on more than four thousand basal vocabulary words. These words could be easily accessed by typing a student's current level, unit, and story. Additional software was designed to improve other areas of weakness as indicated by criterion and mastery tests. Software development was accomplished during the first year of the program and will continue to be an on-going process as more teachers become involved.

COMPUTER ORIENTATION FOR ELEMENTARY STUDENTS, GRADES 4-5

To determine the extent of computer awareness, the students, prior to seeing the hardware, were asked to draw pictures of a computer; most were able to produce a reasonable likeness. A pre-computer attitude survey was admininstered and the results were that most students:

1. had seen a microcomputer
2. had not used one
3. had wanted to learn how
4. thought it would be difficult
5. thought more men than women used computers and children rarely did
6. would not share a computer experience with a friend or

family member because of a fear of failure
7. half of the students felt the computer would not help them with reading because they associated computers with games

On the post questionnaire the main differences were that students:

1. found it easy to learn to use a microcomputer
2. thought men, women, boys and girls all use computers
3. had shared their computer experiences with friends, family and even pets
4. felt they had all received help with reading through computer assisted instruction especially with vocabulary

Before actually using the microcomputer, the students received a list of procedures and rules which were developed to ensure the proper use of hardware. Next, students were given the opportunity to practice on an electric typewriter in order to learn the proper position and touch on the keys. They learned to type their names, the numbers 0-9, and the words "yes" and "no" since most programs require this information.

Students were given a pre-test to determine that vocabulary words used in the initial programs were words which they could read. Thus, they could experience immediate success. Shortly after using the computers, the students developed such an attachment that they perceived the computers as friends and wanted to name them. They sometimes talked to them, argued with them, hugged them, confided in them, and on occasion would even kiss them good-bye.

COMPUTER ORIENTATION FOR MIDDLE SCHOOL STUDENTS

The training of middle school students was approached differently. A cursory introduction to the computer and its parts, including the keyboard, was given. An immediate hands-on approach followed as the teacher provided small group and individual demonstrations. Enthusiasm spread rapidly because students were directly involved in the learning process.

As new programs were made available, the teacher continued to demonstrate to small groups the proper procedures for accessing programs.

Within a two week period students showed remarkable independence in choosing their own programs, in storing the disks, in assisting other students in the class, and in working as peer tutors with visiting students.

Students previously experiencing low self concepts and lacking motivation showed tremendous enthusiasm and self confidence as the computer program progressed. The remedial reading program, to which these students were assigned, had not been looked upon favorably by other students in the school. As the computer program developed, these students were knocking on the door and trying to find out how they could be admitted. The program, in fact, received recognition throughout the county thus allowing students additional opportunites for growth which they had never before experienced.

Now IS the time for all good reading teachers to discover the wonders of microcomputers for reinforcing reading skills.

Diana Lagotic is the Title I Reading Coordinator for the School Board of Alachua County, Gainesville, Florida

Lorraine Losch is an elementary teacher for the School Board of Alachua County, Gainesville, Florida

Lynn Domenech is presently on leave from Alachua Schools. She is president of Teacher Support Software, Inc., Gainesville, Florida

Evaluation Research - A Catalyst for Change in Computing

L.E. Lee D.G. Haughey
Manitoba Department of Education

This paper reports on the evaluation of the Manitoba High School Computer Network which was conducted by staff of the Research Branch, Manitoba Department of Education. The need for the study arose because the contract for computing services was about to expire. This provided the Department with an opportunity to review the existing means of providing service and to explore alternatives. This study acted as a catalyst to generate interest in alternatives.

Introduction

This paper is abstracted from a study of the Manitoba High School Computer Network (Network). The study was to include an assessment of the existing situation, as well as to provide some options for consideration by Departmental management.

The Manitoba High School Computer Network began in September 1973 with 23 batch terminals located in 23 different schools. In 1977 a five-year contract was entered into between the Department of Education and Cybershare Ltd. for the provision of computer services to a maximum of 40 schools. During the 1979/80 school year, 36 schools participated directly in the Network, while others participated indirectly by using terminals in Network schools or by sending student programs to the Department of Education for processing.

Since the Network was established its objectives have been:

1) "to teach computer science and data processing courses;
2) to provide supportive services which will facilitate improvement in efficiency of operation and level of educational performance by students; and,
3) to provide in-school administrative services."

Determination of Network Inputs and Outputs

Previous evaluations of the Network were conducted on the basis of gross costs; (the cost of all services/benefits available through the Network versus all outputs). A problem with this type of analysis was the difficulty of combining the overall cost data with utilization data and attitudinal data in a satisfactory manner for decision making. Thus, a key component of the study was to account, after the fact, for the cost of each service, or set of services, responsible for one of the objectives of the Network. For the purposes of this study, the three aspects to be costed were: in-school administrative services; support of the teaching of computer-science and data processing courses; and, support to courses that were non-computer in content.

The total cost of providing the spectrum of services available through the Network was comprised of fixed costs (those that did not vary with the amount of the service) and variable costs (those that did vary with usage). These costs were shared in varying proportions by the Department of Education at the provincial level and the local school division (district). An assumption was made that the number of System Resource Units (SRU's) best represented the means of allocating the costs to the Network objectives. Data were collected from the operating system on the mainframe which indicated the number of SRU's used by objective. The resulting set of ratios was used to allocate the remainder of the costs.

Data were gathered from administrative records on: the in-school administrative services used; number of students enrolled in computer science and data processing courses; and, the number of programs accessed for use in non-computer courses. Data were also collected on alternatives to the Network used by other schools to deliver comparable services.

All principals involved with the Network were interviewed to obtain their perceptions of the administrative services offered. Teachers of computer science and data processing courses were surveyed to obtain information regarding programing languages taught, satisfaction with various aspects of the Network, and any other computing services being utilized. In addition, students who had successfully completed one or more computer-related courses were surveyed to determine their satisfaction with the course content and Network facilities. As well, these students were asked about their current occupation and whether they had continued their studies in a computer-related area.

A random sample of teachers of non-computer-related courses were surveyed to determine: if they had ever utilized the support programs available; their reasons for utilization; and, their satisfaction with the service. As well, teachers participating in a computer assisted learning (CAL) project for special needs students were surveyed regarding utilization, satisfaction and various aspects of program utilization. Parents of the students participating in this special project were also surveyed to gain an indication of parental awareness of and attitudes towards the project.

Major Findings

Objective 1: to teach Computer Science and Data Processing Courses

- Eleven Network schools (31%) supplemented the Network with a micro computer.
- Enrolment in Computer Science and Data Processing courses varied from 3% to 49% of the grade X to XII enrolment.
- One quarter of the Network schools (9) had an introductory computer course which involved 381 one credit student equivalents, while one Non-Network school had 100 one credit student equivalents in an introductory course.
- Ninety-five percent of teachers from Network schools indicated that they were "somewhat satisfied" or "very satisfied" when asked to comment on "the suitability of the Network to my needs".
- A two course sequence is available in both Computer Science and Data Processing. Between 30% - 40% of students completed the sequence.
- About 50% of the students who had continued their education after high school indicated they had taken further computer courses.
- Students who took further study in a computer course indicated that their high school courses had prepared them "to some extent" or "to a great extent".
- About 15% of the respondents indicated they were employed full-time in a job related to

their computer course. Of these, about 25% came from a non-network school. Fifty percent had attended post-secondary education.

- BASIC was the computer language used most on the job by students who had a computer-related job. Cobol and Fortran were used the least, although they had been taught most often.
- Respondents showed more dissatisfaction with aspects of their course that were equipment related - i.e. availability and reliability of equipment and the type of equipment used.

Objective 2: to provide supportive services which will facilitate improvement in efficiency of operation and level of educational performance by students

2:01 Support to non-computer courses

- Sixty-two percent of the teachers who responded were not aware of programs for use in non-computer courses, while less than 20% had utilized such programs.
- Ninety-five percent (150) of the teachers using the programs stated that they had served their purpose "moderately" or "very well."
- The programs were most often used to "motivate students."

2:02 Computer Assisted Learning (CAL) Project for Special Needs Children and Adolescents

- The cost to operate the CAL Project averaged $4411 per school.
- A positive relationship was found between the increase in the amount of student contact time and teacher perceptions of effectiveness. Teachers who believed their students needed more contact time wanted minimum weekly contact time ranging from 20 to 100 minutes per student.
- Concerning the primary useage, 75% of teachers indicated that "primarily" they used the computer to "drill students on skills already taught."
- Overall, 60% of teachers felt the computer was "somewhat effective" as a teaching tool. Teachers who had experienced some type of specialized training were most likely to regard it as "very effective."
- Eighty-four percent of parents knew their child worked with a computer at school.
- Seventy percent of parents indicated their child's school work had improved, while 36% perceived an attitudinal change.
- When all parents were asked "Do you think students benefit from using a computer", 56% replied "yes". However, a sizeable percentage (41%) responded that they did not know.

Objective 3: to provide in-school administrative services

- Among the 46 schools responding, 50% of the 34 that used a computer to perform administrative services, used the Network Administrative Services Package.
- Schools using the Network Administrative Services Package were generally satisfied.

- Per pupil costs for schools using student scheduling, grade reporting and cumulative record functions on the Network ranged from a low of $7.08 to a high of $26.38. These costs included fixed and variable computer costs as well as materials.

Conclusions/Implications

It was concluded that generally the Network achieved its objective of "Supporting the teaching of Computer Science and Data Processing courses in the high school." However, the following implications surfaced:

- Although the objective is universal, not all schools teaching computer courses used the Network. Furthermore, a pilot vocational program conducted under the auspices of the Department of Education does not use the Network.

- Are computer courses designed to provide computer literacy/awareness or job entry level (vocational) programing skills?

- Elsewhere general education with respect to computers involves integration of computers into other subject areas.

- Is the service cost-effective, given the range of cost per pupil?

It was concluded that students generally did not take a full sequence of either Computer Science or Data Processing courses. Furthermore, it was concluded that training in these courses did not lead to placement in a computer-related occupation. However, the following implications surfaced:

- If the objective of the program is to have students complete a computer course sequence then the objective has not been met.

- If the objective of the program is to provide exposure to students then:
  - The objective has not been met because of the low participation rate; and the program does not demonstrate cost-effectiveness on an overall basis.
  - How much "computer power" is necessary for this purpose?

It was concluded that the Network did not achieve its objective of: "Providing support to other school services, especially those services which will help improve the efficiency of school operation and student performance" as measured by the support to non-computer courses. The following implication surfaced:

- If it is to meet its potential of reaching 100% of students in Network schools then:
  - Teacher awareness will have to be increased beyond current levels; and programs will have to be made available for all subjects.

It was concluded that the Network achieved its objective of: "Providing support to other school services, especially those services which will help improve the efficiency of school operation and student performance" as measured by CAL. The following implications surfaced:

- As the Network was intended for secondary schools and as the majority of students participating in CAL were elementary students, then the objective has not been met.

Notwithstanding the above:

- The CAL Project could be continued independent of the Network.
- Micro computers are a viable alternative to the Network as evidenced by the recent expansion of the CAL Project using a micro computer.

It was concluded that the Network achieved its objective of: "Providing School Administrative Services" on an overall basis. However, the following implications surfaced:

- Is the service cost-effective, given:
  - The decision of the largest urban school division to operate its own system and another urban school division to buy the services elsewhere; and,
  - The quoted prices from private service bureaus.
- Is there a rationale for Departmental subsidy for this service, given the availability of alternatives?

  Options exist if there is no strong rationale for government subsidy to the Administrative Services Package:

  - Recover all costs from users of the Administrative Service Package.
  - Encourage present users to consider other suppliers.

  Options exist if there is a strong rationale for government subsidy:

  - Continue the existing arrangement and provide support to non-users;
  - Provide grants on a per pupil basis for administrative functions;
  - Develop, distribute, and support software for in-house installations.

It was concluded that alternatives beyond the Network were possible and viable.

The following factors should be considered in reaching any decisions:

- Technological change in computing has been phenomenal since 1973.
- The Department was the initiator in computer education in 1973. Currently, a number of schools are initiating new programs and are acquiring new hardware on their own.
- What is the need for computer education in the 80's and 90's?
- What is an appropriate role for the Department of Education?
- What is the impact of the micro-explosion?
  i.e. How many different types can/should the Department of Education support?
- Can the Network be "soft" rather than "hard?"
- What are the non-instructional computer-related needs of schools?

In conclusion, when determining a course of action for computer education and computer utilization, the following must be considered in structuring the objectives:

WHAT is to be provided? WHY?

to WHOM it is to be provided?

at WHAT COST it is to be provided?

and lastly - HOW it is to be provided?

STOREROOMS OR CLASSROOMS? THE FUTURE OF MICROCOMPUTERS IS IN THE TRAINING OF ELEMENTARY SCHOOL TEACHERS

Dr. Tom Richard
Department of Mathematics and Computer Science, Bemidji State University,
Bemidji, MN 56601

ABSTRACT

With the rapid growth of microcomputer usage over the past few years, the major problem of educating inservice elementary school teachers to utilize these machines has arisen. School administrators and computer coordinators have found that it is not enough to buy a microcomputer and a set of manuals, and then "turn the elementary teacher loose." In order for microcomputers to be used effectively in an elementary classroom (or, for that matter, to be used at all) proper instruction must be given. This presentation will describe one successful workshop which is used to acquaint inservice elementary school teachers with what is available for them in the area of microcomputers. In addition, the "hands-on" approach used in this workshop provides the teacher with the confidence necessary to begin using microcomputers in the classrooms.

The presentation will center around the APPLE II, but will be adaptable for use with any other microcomputer. Special emphasis will be placed on the effective classroom use of a variety of materials for

1. drill and practice;
2. computer simulations;
3. tutorials;
4. teacher aids;

and 5. educational games.

The desired outcome from this workshop is that inexperienced microcomputer users will become both familiar with and, comfortable using, microcomputers and, thus, be willing to use this valuable educational tool in their classrooms. Without proper instructional techniques for classroom teachers the future of microcomputers will be in the storerooms of schools rather than in the hands of children.

During the past few years the state of Minnesota has become a leader in microcomputer based instruction at the elementary level. The Minnesota Education Computer Consortium (MECC) has been actively involved in developing excellent software for use in the state's elementary schools. School districts throughout the state have been very supportive in purchasing microcomputers for classroom use. It has been estimated that over 1600 microcomputers are being used in the elementary school classrooms of the state. Of these, 1100 are APPLES and 500 are PETS. With the recent signing of a contract to provide ATARI 400 microcomputers to state school districts at a very reasonable cost, the number of microcomputers will surely increase. With this dramatic increase in the availability of microcomputers, several questions deserve to be asked:

1. "Are our elementary school teachers familiar enough with microcomputers to put them to use in their classrooms?"
2. "Are elementary school teachers aware of the wide range of instructional software available for microcomputers?"
3. "To what use are classroom microcomputers currently being put?"
4. "What is currently being done to provide elementary school teachers with background on how to effectively use microcomputers in their classrooms?"
5. "What needs to be done to insure that microcomputers will become a valuable instructional tool in our elementary school classrooms?"

In conjunction with the National Science Foundation, a questionnaire was used to survey the elementary school teachers in northern Minnesota concerning these and other questions. The results of this survey indicate that instruction on how microcomputers function, on what software is available, and on how to effectively use these machines in the classroom is needed. It was found that without proper instruction for classroom teachers "the future of microcomputers will be in the storerooms of schools rather than in the hands of children." School administrators and computer coordinators have found that it is not enough to purchase a microcomputer and a set of manuals, and then "turn the elementary school teacher loose." In order for microcomputers to be an integral part of elementary school classrooms, proper instruction in their use must be given to the classroom teacher.

To meet this need in northern Minnesota a one graduate credit class was developed and has subsequently been offered fourteen times to elementary school teachers. Over three hundred and fifty (350) inservice teachers have participated in this class with the resulting effect of increased usage of microcomputers in the elementary classrooms. The remaining part of this paper will describe the major components of this class.

Although the class has been offered in a variety of formats and settings, the most typical situation would be for the instructor to travel to the local schools. The average class size would be approximately 20-25 with a desired ratio of at least 1 microcomputer for each 4 students. Although the background and experiences of the participants vary, in most instances they have little or no experience with microcomputers. Since most participants are actively teaching during the week, the class meets from 6:00 to 10:00 on a Friday night and from 8:30 to 4:30 on Saturday. Although a variety of microcomputers have been used, the most commonly used machine has been the APPLE II.

The major objectives of the program are to insure:

1. That participants obtain sufficient hands-on experience to build confidence in their ability to use a microcomputer in a variety of classroom settings;

and 2. That participants be exposed to a wide variety of microcomputer software and be encouraged to select a given program or set of programs to introduce to their classroom.

In order to insure that the first objective is met, approximately one-half of the class time is spent running selected programs. The usual approach is for the instructor to introduce a selected set of programs, perhaps drill-and-practice, and then provide enough time for the students to properly run all of the programs in that area. During the presentation of the programs extreme care must be taken to point out how the program should be used in their classroom setting. In addition, it is necessary to discuss the effective use of the software package, the program's strengths and weaknesses, and the appropriate grade level for each program. Since the majority of the programs presented are APPLE programs developed by MECC, excellent documentation with supporting lesson plans is available. Sufficient time must be allowed to cover this important aspect.

The workshop concentrates on five major software areas which are vital to instructional computing. These are:

1. Drill and practice;
2. Tutorials;
3. Simulations;
4. Teacher utilities;

and 5. Educational games.

The APPLE II microcomputer programs presented in each of the software areas (unless noted) are those developed by MECC for Minnesota's classroom teachers. These programs are readily available to elementary teachers throughout the state and, consequently, receive the most attention. Two additional points should be noted:

1. The programs which are listed are also available outside the state of Minnesota and can be purchased from MECC.
2. Programs similar to the ones listed are available for most all types of microcomputers and, thus, a similar workshop can be developed using the PET, TRS-80, etc.

In the area of DRILL AND PRACTICE the following programs were selected to be run by the participants:

MATHEMATICS: Speed Drill
Tens
Round
Metric Length
Change
Milliken Mathematics (Milliken Company)

LANGUAGE ARTS: Spell
Mixup
Spelling (Primary)
Special Needs Spelling
Phonet (An English Usage D&P)

MUSIC: Introduction
Name the Note
Counting
Missing Note
Wrong Note

SOCIAL STUDIES: States
States 2

PRESCHOOL: Colors (Student Written Program)
Counting Bee (Eduware Company)

It should be noted that this list represents a variety of subject areas and several programs are chosen in each category. This allows for both individual teacher interests and needs.

In the area of TUTORIALS several excellent example programs are available. During class discussion the most commonly illustrated programs are the MECC program FISH and the student written program ENGINE. Both of these programs provide the teacher with the idea of what a computer tutorial is and how one can be effectively used in a classroom setting. The software packages considered include:

MATHEMATICS: Metric Estimate
Estimate
Aestheometry (Curve Stitching)

SCIENCE: Fish
Minerals
Solar Distance

LANGUAGE ARTS: Talk
Prefixes
Kevin's English

MISCELLANEOUS: Engine (Student Written Program)

The programs available in the area of SIMULATIONS are both large in number and rich in experiences. These programs are among those most enjoyed by students, but care must be taken to insure that the students recognize the value of the lesson being presented and not treat the program as a game. Programs in this area include:

SOCIAL STUDIES: Oregon
Nomad
Furs
Sumeria
Voyageur
Civil

BUSINESS/SOCIAL STUDIES:
Sell Apples
Sell Bicycles
Sell Plants
Sell Lemonade

SCIENCE: Odell Lake
Odell Woods
Ursa
Quakes

In order to make maximum use of software in the area of TEACHER UTILITIES, a printer should be available with your microcomputer system. This allows the teacher to generate some valuable classroom materials. Programs in this area include:

TEACHER UTILITIES: Word Find
Crossword
Block Letter
Posters
Review
Test Generator
Percent

The final area considered is that of EDUCATIONAL GAMES. This area should be differentiated from that of COMPUTER GAMES. Although computer games such as SPACE INVADERS are a fun and interesting use of microcomputers, it is my opinion that they are very much overused in the elementary classroom. As illustrated above, there is such a wide variety of educational software available to the trained elementary school teacher that reliance on computer games for the classroom is an injustice. Games to be played on microcomputers during school hours must have educational value and this value must be clearly established with the student user. If we are going to allow our students to spend instructional time on microcomputer games with little or no educational value, the place for microcomputers should be in our schools' storerooms. Do not waste valuable instructional time!

With the above note of caution in mind, several games are presented to the class participants for use with their students. Although the following list does not exhaust the list of good game software, it includes:

EDUCATIONAL GAMES: Metric 21
Hurkle
Taxman
Bagels
Number
Math Game
Word Game

In summary, the workshop described above is used to acquaint the inexperienced microcomputer user with proper instructional use of the APPLE II. Great care must be given to see that neither elementary school teachers nor elementary school students are made to "fend for themselves" with microcomputers. Teachers must be shown the wide variety of software available for the microcomputers of their choice and proper instruction must be given on how to get the most educational value from their programs. Without proper instructional techniques for classroom teachers the future of microcomputers will be in the storerooms of schools rather than in the hands of children!

THE NUTS AND BOLTS OF SELECTING AND MAXIMIZING COMPUTER ASSISTED INSTRUCTION

Richard D. Conkling, Principal
Bunnell Elementary School, Flagler
County School Board
Ed.D. University of Miami, Coral Gables

"So profoundly revolutionary is this new civilization that it challenges all our old assumptions. Old ways of thinking, old formulas, dogmas and ideologies, no matter how cherished or how useful in the past, no longer fit the facts. The world that is fast emerging from the clash of new values and technologies, new geopolitical relationships, new life styles and modes of communication, demand wholly new ideas and analogies, classifications and concepts. We cannot cram the embryonic world of tomorrow into yesterday's conventional cubby holes." Alvin Toffler, The Third Wave.

Alvin Toffler was speaking of the new era which he refers to as the Third Wave of civilization. The first wave being agricultural and the second, industrial. While no one thing is attributed to the creation of this new wave, technological developments in communications and computers are key ingredients. Accordingly, Toffler suggests that "Computers are enhancing our brain power much the same as machines enhanced our muscle power during our industrial era."

The effect computers have had upon our ability to manipulate and manage information is tremendous. Even now we are only just beginning to tap their potential in many areas of manufacturing with programmable robots. Educators are only just beginning to see the potential of computers as educational tools. It is becoming the thing at educational gatherings to talk about the wonderful things we are doing in our schools back home with computers. Computer assisted instruction is now a byword in education.

However, in the rush to enter the arena of computer assisted instruction many districts and schools are leaping in before they have tested the depth of the curriculum or determined their own needs. As a result, computers are being purchased without a clear understanding of how they are or can be used or how they will fit into their instructional program. Thus many computers are ending up as novelties which attract attention but are making little or no contribution to the education of children. Many of those that are being used effectively in an instructional program are only used for brief periods of time by a select group of teachers within a particular school.

Computer assisted instruction can be a definite asset to the instructional program of a school if it is entered into with a planned program to meet established objectives. The simple act of purchasing computer hardware is relatively easy. The purchase of a computer which can provide the software capabilities which can be most effectively used within an instructional program is not so easy.

A school or district must take the time to determine the needs of the population it intends to serve, the size of the population and the manner in which the students could be best served. A school may only have a single objective in mind or the objectives may be very diverse, seeking to fill some very divergent needs. Whatever the objectives are, they should be the only limiting factors in the initial review of the capabilities of various computers and their programming to serve your needs.

You cannot assess the potential of of computers as instructional tools until you are aware of all the possibilities that are available to you. While in the final analysis you may not be able to afford the very best for your situation, it will provide you with a measure to compare others. An effort should be made to view all possibilities offered by micro-computers, mini-computers, large computers, or various time-sharing configurations. Too often we have a fixation on a type or style of computer we want to review and overlook others.

Good computer assisted instruction does not come free and financial resources must be considered. However funds are too often used as a limiting factor similar to preset notions about a certain type of computer. Operations which can potentially better serve your needs may be bypassed when in fact its cost may not be an insurmountable obstacle. School districts often have the opportunity to enter into time-sharing arrangements, lease to purchase agreements, pooling of resources agreements or private industry support programs. It is surprising sometimes how funds can be generated to enter inovative programs when someone becomes very determined and convinced of the need.

A critical factor in the use of computer assisted instruction is the maximization of the computers once they have been purchased. Sometimes we are led to believe that microcomputers are so inexpensive that they can just be scattered around through schools to be used whenever the occasion arises. As many of us have found that inexpensive $700 computer ends up costing nearer to $3,000 if it is to have the necessary capabilities to serve the needs of a well developed computer program.

It is essential that the usage of our computers be well planned. Computers can be put on wheels and moved to students or students can be moved to the computers. But there are many other questions which should be addressed. Who will be responsible for the hardware? Who will be responsible for the software? Does someone have to be trained to write programs or adapt programs? Where will the computer be housed? Where will the soft ware be stored? How many students do you plan to serve? How much time will the students spend "on" the computer? How often will the students have an opportunity to be "on" the computer? Does the software have built in time constraints? Can the time constraints be adjusted? How many hours a day can students be scheduled to use the computer?

Nearly every school board will either ask or should ask whether or not computer assisted instruction is cost effective. Cost effectiveness of a particular program is difficult to ascertain and even more difficult to evaluate in a normal school setting. Even though such evaluations may not be perfect, a reasonable attempt should be made. Cost comparisons can be made between programs along with a comparison of achievement gains. Comparisons can also be made between cost of gain prior to the introduction of the new program and the cost associated with any additional gain which could be attributed to the new program. One important factor that is too often overlooked is that computers may not be the most effective way to achieve the results you desire.

The Flagler County School system in North East Florida entered into computer assisted instruction in the fall of 1980. The system is relatively small, serving approximately 1900 students. A brief summary of its entrance into and application of computer assisted instruction may help to illustrate the use of many of the suggestions just made.

In 1978 Flagler County began to investigate the possibilities of using computers to assist in its instructional program. The process began by establishing a series of objectives to lend some direction to the investigation. A series of five objectives were informally developed. (1) Look for an instructional program that would provide a stimulus to instruction and maximize its effectiveness. (2) Provide a program for students who need remediation. (3) Find a program that can be used by highly motivated students. (4) Look for a program that could be used by elementary, junior high and senior high students. (5) Find a program than can provide individualized and diagnostic instruction.

There is no doubt that the objectives were ambitious but they did provide direction. Fortunately it was decided at the beginning of the investigation to forget cost until after the investigation of available computer assisted or computer managed instructional programs was completed. As a result we were able to view a whole range of possibilities and make intelligent comparisons. The instructional program which could best meet our objectives was a mini-computer with a series of terminals. A contributing factor to this decision was a limitation of a small districts ability to provide personnel with the expertise needed for programming computers to adapt instructional programs or to develop them inhouse. The particular computer and program purchased provided its own unique curriculum.

In the investigation we encountered problems which are still in existence today even though in some cases to a much lesser degree. The problems were a lack of people with experience using computer assisted instruction on a large scale within our area, no central source concerning availability and potential of various programs or computers, the difficulty in evaluating computer curriculums and the aura about computers which tends to mask the true merits of programs or equipment.

Mini-computers are considerably more expensive than micro-computers and require special consideration when it comes to financing. Several districts have entered into operations similar to the one in Flagler County through the use of Title I funds or other special large scale projects. In a large district, such costs can be absorbed by special projects and the program can be used to serve that select population. However, in a small district with only limited funds in any single funding source and a limited number of students to be served within them, it becomes imperative that such financing and use be distributed according to the need. While such an idea sounds logical, it is very difficult to accomplish with the restrictions concerning co-mingling of funds in federal projects.

This was a major problem in Flagler County. After the investigation, there was no question about which computer set up would most adequately serve the needs of the various segments of the district. The main problem was finding a way to pool funds together to finance the project. A private corporation which was building a large development within the district had expressed interest in providing some financial support for an innovative instructional program in the schools. One of the schools in the district was receiving Title I funds and the other school was receiving State Compensatory Education funds. Also there was a limited availability of funds from the general school fund.

The computer project was financed by entering into a lease-to-purchase agreement which helped to spread out the cost over three years. The private corporation made a committment of funds to support the project over the same years. Title I funds and Compensatory Education funds would be used to support the program however none of their funds could be used to purchase the equipment. Finally, local funds were used to make up the balance necessary for the purchase. To be very honest, it took a lot of hard sell to gain school board approval but by establishing objectives in the very beginning, the ground work was established for justification of the program. In the final analysis the program was only approved for one year after which district personnel were required to demonstrate that the program was able to show gains in academic achievement.

Flagler County purchased a computer program which was able to meet almost every one of its initial objectives. The problem then became one of maximizing it use as an instructional tool not only for an individual student but for a maximum number of students. The computer assisted instructional program is essentially the same in manner of operation at both the high school and elementary school. Therefore only the elementary school program will be discussed here.

The primary target at Bunnell Elementary School is 144 Title I students in grades four, five and six. These students come in groups of 24 to a Title I room for 40 minute periods. During that period 3 groups of 8 students each will be served ten minutes each by the computer. Each student signs on into his own particular program and the computer picks up where the student left off the previous day. The program adjusts forward, backward, faster, slower, provides branching and some tutorial assistance. After 10 minutes or the assigned amount of time for the student, the terminal will automatically sign the student off. This format applies to the major programs in reading and mathematics.

Approximately 144 Title I students are served in this manner every day on the reading program. However during the 30 minute lunch period for Title I, a third grade class brings in 16 students each day and runs two 10 minute sessions in computational math. Every morning during the school's opening exercises, 8 additional students are served as are another 8 during the last 15 minutes of the school day when Title I is not in operation. Every afternoon immediately after school, 8 more students come in and work 25 minutes and are picked up by their parents. In total at the elementary school 184 students are served each day in a planned computer assisted instructional program. This represents 17% of the student body in the K-6 elementary school. It represents 30 hours of online computer assisted instruction 5 days a week for 170 days during the regular school year.

The Computer Curriculum Corporations computer assisted instructional program has been in existence in Flagler County since September of 1980. At the end of its first year an evaluation of the program was made by asking a series of questions.

How effective was it educationally?

The Comprehensive Test of Basic Skills is administered yearly to students in Flagler County Schools. A comparison was made of the grade equivalency achievement gain in mathematics for 83 students who participated in the Title I or the State Compensatory Education program during the year immediately preceding the introduction of computer assisted instruction and the year following with it. These students gained an average of 4.3 months more the year they were served with computer assisted instruction.

Does computer assisted instruction stimulate students?

Students who were served by the computers were just as eager to sit down at the terminal and sign on in May as they were in September. Also students who participated in the computer programs were asked to respond to an attitude survey which asked questions about their total school instructional curriculum. Every question which involved their participation on the computers had a much higher positive average than the other questions.

Is computer assisted instruction cost effective in Flagler County?

Computation which compared cost of gain in the regular program versus the cost of additional gain received from the computer program indicated that the additional gain cost approximately $11.00 less per month of gain.

How many students were served by the 16 terminals provided in the program?

A total of 347 students participated on a daily basis in the computer instructional program. An additional 46 students were served with it at other times in enrichment programs, adult education, exceptional education, and language arts programs.

In summary the computer assisted instructional program has been an effective educational tool in Flagler County. A program was selected to meet identified objectives which were identified within the county. In order for computer assisted instruction to be effective it should be a planned part of the instructional program.

Toffler, Alvin. The Third Wave.
New York:Morrow & Co. 1980

APPLYING COMPUTERS EFFECTIVELY IN THE SMALL TWO_YEAR COLLEGE

Lora P. Conrad

Director of Development and
Computer Services at Northwest
Alabama State Junior College

Louis W. Bender

Director of Institute for
Studies in Higher Education
at Florida State University

INTRODUCTION

Small colleges nationwide are adding or upgrading information systems and computer centers--posing new challenges both management and fiscal for their presidents and administrators. Information on the issues faced by small colleges in procuring, managing, financing and effectively using computers and information systems is largerly unavailable, as the literature documents primarily the concerns and status in larger colleges. Seeking such information, research was conducted for the purpose of defining the state-of-the-art of computers in small colleges, determining factors that contribute to success in meeting the challenges of applying information systems effectively, and identifying common concerns.

THE STUDY

Case studies were conducted of five small (less than 2,000 credit headcount), two-year public colleges. The procedures for selecting the colleges to study began by contacting the state education departments in fourteen southern region states to identify those small colleges believed to be most advanced in their use of computers and information systems. From the resultant group, five were selected for study to provide one college each from five states with varied approaches to meeting computing needs, size variation, and differing degrees of state level involvement. In-depth profiles were developed through on-site interviews, observation, and review of documentation, followed by compilation and synthesis of information.

Research questions were answered concerning policies, governance, organization, systems planning, systems procurement, hardware used, software capabilities, internal procedures, costs, user interface, problems, and future directions.

RESULTS

Space does not permit reporting all results, but some of the observations made are conclusions reached about computers and information systems in the small college are presented in the following paragraphs.

Planning and Procurement. Though long-range planning for hardware upgrading and information systems development is rarely conducted by small colleges, its effectiveness is clearly seen among the more outstanding small colleges.

State level policies and procedures that support colleges in planning and procuring information systems and computer hardware have a positive effect on the resultant systems in small colleges. Combined expertise from the colleges and the state can lead to more effective procurement benchmark standards, and procedures than when a college must work alone.

A college facing planning and procurement decisions without strong state support can find effective assistance in Directors of Computing from other colleges or non-vendor consultants knowledgeable about education.

Hardware. The predominant means of accessing computer capability is an in-house minicomputer with state or university shared systems the next most common method. In-house hardware systems most effectively utilized are manufactured by top names in the minicomputer field, with IBM and DEC the most frequent hardware suppliers.

Software. Small colleges have utilized neither software consultants nor proprietary software extensively. Their primary sources of information systems are in-house development, other in-state colleges and state provided software.

With software development costs now exceeding hardware costs in the data processing industry, proprietary software or software from other colleges should be used more extensively and is now being considered to replace some in-house developed software at the institutions studied.

Administrative information systems capabilities are most extensive in Student Records and Financial Systems, beyond that the applications developed are highly varied by institution and may include Academic Support, Financial Aid, Inventories, Student Tracking, and profiles for research and management.

Extensive, integrated management information systems are rare. Locally developed systems are usually day-to-day, control level. State provided systems frequently lack an orientation to the needs of the colleges.

Development of information systems is a continual process. The current stage at an institution is more a function of the length of time since the last major hardware (computer) change than the total length of time the college has had a computer center.

Policies. The policy of shared use of one computer center for both instructional and administrative applications is not only expedient but essential in permitting the small college to afford adequate hardware and personnel for either purpose. However, this is contrary to recommendations frequently seen for larger colleges.

The dual role of the computer center is also reflected in a variety of governance issues that result, including a dual instructional/administrative role for the Director of Computing. This is not easily reconciled with the organization structure of most small colleges and thus leads to a high degree of autonomy for the astute Director of Computing.

Presidents that understand well the power of information strongly support the computer center and development of information systems through helping "sell" the concepts internally, budgeting for required system upgrade costs, and maintaining direct interface with the Director of Computing. Presidents rarely utilize directly the reports available.

Users. The availability of sophisticated analysis or simulation tools, either state provided or resulting from a sophisticated level of expertise in the Director of Computing, does not mean that users can or will utilize such tools nor information. Users frequently do not fully utilize the information systems capabilities available to them due to factors that include fear of such systems, inadequate training, lack of confidence, or lack of recognition of the value of such capability.

The quality of information systems, computer operations, and computer center staff has little effect on the use of computers in non-data processing areas of instruction. In fact, other uses of computers such as computer assisted or managed instruction are not extensive at small colleges, even those with exemplary information systems.

Costs. Annual operating costs for an effective center range from 2.5% to 6.4% of educational and general expenditures, with in-house minicomputer systems capability far more expensive to the institution than use of a state shared system.

Hardware purchases were primarily financed through state or institutional funds. Local community funds were also a major factor, but very little hardware was purchased with federal money.

Problems. Insufficient funding to upgrade hardware as needed for technological currency, communications between users and computer center staff, turnover of both computer center and key user office personnel were all identified as problem areas.

Directions. Computer applications to administration, instruction in data processing, and as an instructional tool increase in the small college each year, though they continue to lag behind the larger institutions.

Word processing is just now being introduced with the same dual need for administration and instruction. Paperless office concepts are still unknown and will remain cost prohibitive. Users readily adopt increased sophistication at the data entry level but do not readily accept simulations or projection tools.

SUMMARY

The status of information systems in small two-year colleges is generally developmental and unsophisticated though some institutions are effectively utilizing their hardware and information systems. State leadership in planning, procedures guidance, and procurement assistance for hardware and software can be highly effective. Small college leadership perceives their own in-house minicomputer, especially if usable as part of a larger network outside the college, as almost the only way to effectively meet both instructional and administrative computing needs on their restricted budgets. The instructional emphasis is on data processing rather than the computer as an instructional tool. User attitudes are typical of any organization in which computers are being introduced, resulting in highly varied levels of use of the available information systems, though user training is rarely provided. Costs of operating an in-house minicomputer center at a small college are higher as a percent of the budget than at larger institutions.

Effective use of computers in the small college requires a president with an understanding of the power of information and a willingness to pay for it, a Director of Computing with an extensive analysis background, an understanding of education, and a tolerance for users. Users attitudes and utilization of the computerized data base rather than maintenance of duplicate manual records is crucial to data base validity.

BIBLIOGRAPHY

Dr. Lora P. Conrad

Developed information systems and the computer instructional program at Northwest; consultant on information systems and educational applications of computers to community colleges and secondary schools; previously in industry, on Apollo and Shuttle simulation teams, SAFEGUARD MIS.

Dr. Louis W. Bender

Professor of Higher Education at Florida State University, formerly served as Assistant Commissioner of Higher Education in Pennsylvania, and Past Chairman of the National Council of State Directors of Community Junior Colleges. He has authored numerous articles and monographs and co-authored two books; and consultant in the development of eight state master plans for postsecondary education as well as consultant to institutions and organizations in every state.

212

INTEGRATING CAI AND VIDEOTAPE FOR SCIENCE AND MATHEMATICS

Mila E. Ignatz
Research Coordinator & Associate Professor
Florida A & M University Laboratory Schooo
Tallahassee, Florida

Microcomputer technology is interactive in delivery. Where other instructional aids are passive and linear in format, computer-assisted instruction provides active and non-linear instruction. Branching instructions allow an instructional program to be tailored according to the levels of mastery of individual students. Students can proceed according to their own rate of learning; thus, students are on task most of the time, if not all the time. Furthermore, the microcomputer can automatically score and tabulate skills successfully mastered as well as areas of difficulty.

The microcomputer-videotape combination permits the development of an even broader range of instructional activities never before possible with microcomputers alone. Instructional materials can appear on the screen alternately from the microcomputer or the videocassette recorder/player depending upon what is needed as designed in the instructional program. Therefore, moving, colorful, and visual materials as well as spoken descriptions, instructions and music can now be added to the more formal, text-bound character of computer-assisted instruction.

Possible modes for integrated CAI-videotape presentation of instructional materials include: (1) the combination of any two of the following--computer display, video display and videotape sound; (2) branching forward or backward to frames within the videotape; (3) frozen-frame video display while the video player is on pause; and (4) replay of a previous section of video.

The integration of the computer and the video player offers several advantages including the following:

(1) Videotapes of actual lectures, demonstrations, directions, explanations and experiments by teachers, lecturers, and university or college faculty may by incorporated into CAI lessons;

(2) With appropriate branching instructions, the instructional materials with the same data base can be modified to utilize different learning strategies while presenting materials to students with different levels of achievement;

(3) The teacher can conveniently group students without having to repeat lessons and explanations to each group;

(4) Groups of students can interact with the video-CAI program whenever they need to go over them for review or notetaking;

(5) Students can study difficult presentations repeatedly;

(6) The same programs may be used by different classes for a number of years;

(7) Tapes and computer programs are easy to edit and revise; thus,

they can be updated more quickly and easily than a textbook and teachers can adapt them to their teaching styles or preferences; and

(8) They are cost-effective.

The process for incorporating videotape presentations into the CAI program is briefly described as follows. The video production is first taped and edited. Then, the videotape is prepared for integration by dubbing audio signals onto the audio channel 1 of the videotape or whichever channel is not used for the normal sound track of the video program. The audio signals that are recorded are the tape identification number and the frame numbers. Finally, frames from the videotape are inserted into appropriate points of the CAI program by means of video control mode commands.

Existing videotapes may be used if they are suitable and adaptable. However, the educational innovator might find himself compelled to develop his/her own videotape materials if the duration, nature and design of the video presentation is expected to play a vital role in facilitating the achievement of the course objectives. Many already available videotapes are expository in nature and do not lend themselves to the problem-solving approach.

A trigonometry lesson on the graphing of sinusoids and a simulation experiment on titration are being planned to be demonstrated via the integrated CAI-videotape technique during the AEDS convention.

REFERENCE

Schwartz, Marc. Integrating CAI & videotape. *Creative Computing*, September 1980, 7 , 116 - 117.

ALGORITHMS AND THEIR SOURCE CODE IMPLEMENTATION

David A. Nuesse

Chairman, Department of Computer Science
University of Wisconsin-Eau Claire

In developing the solution to a problem, a top-down, step-wise refinement approach is used to arrive at the final algorithm (sequence of statements to be performed in order to produce some desired results), and its program implementation.

To begin, an exact statement and/or understanding of the problem to be solved is needed. The initial problem solution is concerned with broad issues and is developed using expressions (from the world of the problem being solved) which are machine and programming language independent -- details of the solution are postponed until later levels. At each level, alternative ways to design the next level by refining one or more parts of the current level are considered; i.e., stepwise refinement. Each level of the problem solution must be correct so that further refinements will yield a final algorithm which is complete, correct, explicit and translatable into a computer language.

The top-down approach taken to the development of the algorithm involves decomposing a problem into a simple sequence of sub-problems. That is, to solve problem P, a set of smaller problems, P1, P2, P3,... are developed such that solving P1 first, then P2, etc., will be equivalent to solving the original problem P. The same approach can be used to solve P1, P2, etc. by developing sub-sub-problems.

To aid the development of the algorithm, a limited number of control structures are used. The control structures are Sequencing, Selection, Indefinite Iteration, Definite Iteration:

SEQUENCING - In sequencing, the logical steps are taken in the order in which they are presented. That is, one starts at the beginning and works towards the end.

SELECTION - Selection provides for conditional branching within our logic. The three forms of selection are *If* C *then* S (Single Alternative), *If* C *then* S_1 *else* S_2 (Double Alternative) and *Case* I *of* S_1, S_2,...,S_k (Multiple Alternative). C is a condition to be tested and is a boolean expression which is *true* or *false*. S is a single statement (or block of statements which can be treated as a single statement) which represents action(s) to be performed as based on the value of the expression C. In the *Case* structure, the value of I will determine which of the S statements to be performed.

ITERATION - Iteration implies a loop - a sequence of actions to be accomplished zero or more times. There are two types of iteration -- indefinite and definite. Indefinite iterations (*While* and *Repeat*) are loops in which the number of passes through the loop is indefinite. Indefinite iteration - The *While* C *do* S construct implies that while the condition C is *true*, the statement(s) S are to be performed. Note that in the *While do* the test is at the top of the loop and if C is false initially, the loop will be entirely avoided.

The indefinite iteration *Repeat* S *until* C implies that the statement(s) S are to be repeated until the condition C becomes *true*. In a *Repeat until* the test is at the bottom of the loop and, thus, statement(s) S will always be carried out at least once.

Definite iterations (*For*) are loops in which the number of passes through the loop is known in advance. In the first form, *For* I ← M *to* N *do* S, the index I is increasing from M to N, with the statement(s) S performed and I increased by one for each pass through the loop. If the value of M exceeds that of N when the loop is initially encountered, the statement(s) S are not performed (the loop is entirely avoided). In the second form, For I ← N downto M do S, the index I is decreasing from N downto M, with the statement(s) S performed and I decreased by one for each pass through the loop. If the value of N is less than that of M when the loop is first encountered, the statement(s) S are not performed (the loop is entirely avoided).

Consider the following problem:

Compute a value for the mathematical constant e, where

$$e = 1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \cdots$$

Assume that an accuracy specification E will be read and used to calculate e.

An initial level of an algorithm for the solution of the problem could be:

```
A0 Read the accuracy specification
   Compute e to desired accuracy
   Print the value for e
   Stop
```

The next level (A_1) represents a refinement of level A_0 and includes the basic looping structure.

```
A1 Let SERIES represent the series sum
       for e
   Let TERM represent a term of the
       series
   Let E represent the accuracy
       specification
   Read E
   Determine if E is valid
   If E is not valid
     then take appropriate action
     otherwise
       SERIES ← 0
       TERM ← 1
       REPEAT
         Add TERM to SERIES
         Compute a new TERM
       Until TERM < E
       PRINT SERIES
   Stop
```

NOTE: The last term computed is less than the accuracy specification E and is not included in the SERIES sum.

In the next level, we use a more formal declaration for the variables, recognizing the four types of values: REAL, INTEGER, BOOLEAN, AND CHARACTER.

```
A2 Declare var SERIES, TERM, E : real
   Declare var NTERM : integer
   Read E
   If E ≤ 0 or E > 1
     then Print "Invalid accuracy
               specification",E
     else SERIES ← 0
          TERM ← 1
          NTERM ← 1
          Repeat
            SERIES ← SERIES + TERM
            TERM ← TERM * (1 / NTERM)
            NTERM ← NTERM + 1
          Until TERM < E
          Print SERIES
   Stop
```

Having refined the algorithm to a level which is complete and detailed, we still have an algorithm which is basically machine and language independent. Now comes the task of implementing the algorithm in source code.

In what follows, the algorithm will be presented in source code using the BASIC, FORTRAN and PASCAL programming languages so that comparisons can be made between the algorithm implementations.

Please note carefully how the If then else and Repeat Until control structures have been coded in BASIC and FORTRAN. The indentation of statements in the algorithm and in the source code is intentional and necessary for determining which statements are a part of each control structure.

```
10 REM-PROBLEM-COMPUTE MATHEMATICAL E
15 REM-BASIC IMPLEMENTATION
20 REM-VARIABLES
30 REM     E-ACCURACY SPECIFICATION
40 REM     N-TERM NUMBER
50 REM     S-SERIES SUM
60 REM     T-TERM OF SERIES
70 REM
```

```
100 PRINT "ENTER THE ACCURACY ";
105 PRINT "SPECIFICATION ";
110 INPUT E
120 IF E <= 0 THEN 140
130   IF E < 1 THEN 170
140     PRINT "INVALID ACCURACY ";
145     PRINT "SPECIFICATION"
150     GO TO 270
160 REM ELSE
170     S = 0
180     T = 1
190     N = 1
200 REM REPEAT
210       S = S + T
220       T = T * (1 / N)
230       N = N + 1
240     IF T >= E THEN 210
250 REM UNTIL
260     PRINT "MATHEMATICAL E IS", S
270 STOP
280 END
```

Lines 120 and 130 are necessary for testing multiple boolean conditions and appropriate branching for the If then else. Line 150 causes a branch forward to avoid the else portion. Line 240 provides the appropriate test to determine whether to stay in the loop (branch back to statement 210) or exit the loop.

```
C     PROBLEM-COMPUTE MATHEMATICAL E
C     FORTRAN IMPLEMENTATION

      REAL E, SERIES, TERM
      INTEGER NTERM
      READ(5,10)E
   10 FORMAT(F10.8)
      IF(.NOT.((E .LE. 0.0) .OR.
     *  (E .GT. 1))) GO TO 100
C
        WRITE(6,20)E
   20   FORMAT(1X,'INVALID ACCURACY',
               'SPECIFICATION',F10.8)
        GO TO 300
C
  100   SERIES = 0.0
        TERM = 1.0
        NTERM = 1
C       REPEAT
  200     SERIES = SERIES + TERM
          TERM=TERM+(1.0/FLOAT(NTERM))
          NTERM = NTERM + 1
        IF(.NOT.(TERM .LT. E))GO TO 200
C
        WRITE(6,30) SERIES
   30   FORMAT(1X,'MATHEMATICAL E IS',
     *         F12.8)
C
  300 STOP
      END
```

The first IF statement has the boolean expression negated (.NOT.) so as to keep the then portion preceeding the else portion and preserve the actual condition to be tested. The GO TO 100 causes a branch to the else portion. The GO TO 300 causes a branch down to skip the else portion. The second IF statement has a GO TO 200 to branch back to create a loop and uses a .NOT. in order to continue to show the condition being tested. Note that this loop has a test at the bottom of the loop.

```
PROGRAM COMPUTEMATHE(INPUT,OUTPUT);
(*COMPUTE MATHEMATICAL E-PASCAL
  IMPLEMENTATION*)
VAR
  E,SERIES,TERM : REAL;
  NTERM         : INTEGER;
BEGIN
  READ (E)
  IF (E <= 0.0) OR (E > 1.0)
    THEN WRITELN('INVALID ACCURACY ',
          'SPECIFICATION',E:10:8)
    ELSE
      BEGIN
        SERIES := 0.0;
        TERM := 1.0;
        NTERM := 1;
        REPEAT
           SERIES := SERIES + TERM;
           TERM := TERM * (1 / NTERM);
           NTERM := NTERM + 1;
        UNTIL TERM < E;
        WRITELN('MATHEMATICAL E IS',
               SERIES:12:8);
      END;
END.
```

In each implementation the attempt is made to preserve the character of the algorithm so that the programmer can directly compare the logic of the algorithm to that of the source code.

The practical side of algorithm development and source coding suggests that, as the problem becomes complicated (and thus the algorithm), there is a greater chance for the introduction of logical errors. When an error becomes apparent, beginning programmers are often tempted to "debug" the logic by directly modifying the source code. THIS IS A MISTAKE. The modification of source code without prior modification of the algorithm from which the source code was derived usually results in the 'patching' of logic which, in turn, causes the source code to become unreadable and, more often than not, wrong. Experience suggests that the problems of patching are often related to the improper use of the GO TO statement and branching out of loops.

In BASIC and FORTRAN there is a great temptation to misuse the GO TO statement by branching back and forth indiscriminately within the code. To do so is to create code that is very difficult to understand. In fact, the GO TO statement for unconditional branching should be used for only two purposes:

1) Branch forward in order to skip code
2) Branch back in order to create a loop and thus re-execute code.

Loops such as the While, Repeat, and For are designed so that the exit from the loop is at the top or bottom of the loop but never in the middle of the loop. The 'patching' process often results in using a GO TO statement to exit inside the loop, again complicating the logic.

A better approach to the elimination of the errors is to return to the algorithm and find the level which does not contain the logic error. One should proceed (through step-wise refinement) to develop new levels which correct the mistake and, at the same time, preserve the remainder of the correct logic. If in this process a new error is uncovered, back up to an earlier (and correct) level and proceed top-down once more.

The point of this is that the 'debugging' best belongs in an examination of the algorithm, not the source code. The logical design of a problem solution is an algorithm development process.

No matter what programming language is used for the final source code implementation, the code should directly reflect the logic of the algorithm. At the University of Wisconsin-Eau Claire in our CS 135 - Computer Programming in BASIC course, a handout is provided for achieving the desired correspondence between program and algorithm. A similar handout is used in our CS 151 - Computer Programming in FORTRAN course. However, no such suggestions are necessary for our CS 155 - Computer Programming in PASCAL course because the PASCAL language has statements which directly correspond to the control structures of the algorithm.

The use of the PASCAL language in our introductory courses in our computer science major/minor sequence is a distinct improvement over our previous use of other languages. The process of learning to program computers is terribly complicated. We need all the tools we can find to simplify the learning procedure. Problem solvers and programmers always need to understand each of the following:

1) The problem itself
2) The algorithm development process
3) The syntax and semantics of the computer language to be used.

PASCAL, being a language which puts emphasis on the algorithm control structures, is a tool well suited to the learning process, and makes it possible for us to place added emphasis in our classes on the much more difficult algorithm development process. Our students learn more and become better problem solvers because of the use of the PASCAL programming language.

REFERENCES

Wayne Amsbury, Structured BASIC and Beyond, Computer Science Press, Rockville, MD, (1980).

Richard Conway and David Gries, An Introduction to Programming, Winthrop Publishers, Cambridge, MA, (1975).

Roy F. Keller, Constructing Algorithms, Kendall/Hunt Publishing Co., Dubuque, IA, (1975)

Elliot B. Koffman, Problem Solving and Structured Programming in PASCAL, Addison-Wesley Publishing Co., Reading, MA, (1981)

Niklaus Wirth, "Program Development by Stepwise Refinement", Communications of the ACM, Volume 14, Number 4 (April 1971), pages 221-227.

MATCHING INSTRUCTIONAL PROGRAMS TO LEARNING PROBLEMS

James P. Pollard, Ed.D.

While there is still no consensus among professionals as to the meaning of the term "learning disability" or as to which specific learning problems might be found within the syndrome, most of the literature is in agreement upon the need for a complete assessment of these students. The federal regulations which mandate educational services for handicapped students contain specific provisions for assessing students who are thought to be learning disabled. Educators who use computer assisted instruction in their classrooms have a unique advantage when selecting courseware for use by learning disabled students since the selection guidelines are contained in that assessment.

Then evaluation of a learning disabled student is usually conducted for two purposes--to identify the student in response to funding, due process, and other administrative needs; and to provide information to use in placing the student, in deciding on his educational objectives, and in planning remediation strategies (Wallace & Stevens, 1978). The completed assessment will contain the results of a number of formal and informal tests and observations which culminate in the designation of the student as handicapped and in an individualized education program (IEP). In this paper a number of the measures which appear in a student's evaluation will be discussed in relation to selecting courseware which furthers the objectives of that IEP and helps to take the student's disability in mind when selecting materials for his mainstream classes.

PERCEPTUAL TESTS. It is commonly believed that many of the learning disabled difficulties in school can be traced to a specific perceptual problem which interferes with his learning. In reviewing the evaluation for courseware selection, attention should be paid to the type of learning, or modality, to emphasize or avoid. A student's dominant modality is typically described as auditory, visual or motor and may differ for expressive or receptive learning (Reger, Schroeder, & Uschold, 1968). A student who learns best through visual channels and who can express himself through motor channels is an obvious candidate for CAI. Programs which have been designed for use by aurally handicapped students and low reading level students are particularly helpful for these students.

A student who has difficulty with visual perception may benefit from courseware which uses large graphic lettering sets, auditory cues, and motor responses. The student may be given cues in decoding the written material through screen prompts and controlled rate of presentation.

Macleod and Overheu (1980) reported uses of microcomputers in the direct remediation of visual-motor problems. While their hardware modifications may not be available to teachers, programs which use game paddle responses to track screen activities are widely available (see, for example, Space Invaders, Asteroids, etc.) Most arcade games combine good practice in visual-motor integration with excellent motivation and low reading levels.

If a student's individual evaluation does not include a discussion of perceptual acuities, the teacher may discover the student's best modality by reviewing the tasks at which the student fails. If the input mode or the output mode of the task is consistent through a student's failures, the selection of courseware should be to advoid those channels in teaching and toward those channels in remediation.

SKILL LEVELS. The tests of specific skills which are included in an evaluation are useful both in determining the specific special education tasks which the courseware should address and in determining whether the student has the entry skills necessary to complete CAI lessons in any field. In a recent review of special education uses of computers, Lindsey (1981) found only one product which was specifically advertised as appropriate for handicapped students. It should be expected, therefore, that most courseware will assume a reading level similar to the level of the subject matter. If this assumption is not true, the mainstream teachers should be warned of the possible problems in using existing CAI materials.

MEMORY TESTS. All students who are considered learning disabled should have been tested for general intellectual potential. The memory portion of these tests can be helpful in courseware selection in that it indicates how quickly the student should be reinforced after a correct response, how often material should be repeated, how much material can be learned in one session, how often instructions should be reinforced, and how soon after an instructional component a student should be tested. The psychological report should include a discussion of general intelligence and will probably discuss memory. If this topic is not specifically discussed, the teacher should ask the psychometrist for guidance in determining the types of memory which represent particular strengths or weaknesses.

If the student does have a memory deficit, the use of drill and practice CAI materials has been proven effective for remedial mathematics (Modisett, 1980) and reading (Jamison, Suppes & Wells, 1974). While the documentation for drill and practice courseware has been inadequate (Jostad & Kosel, 1980), teachers should look for those products which allow close control over the content and format of the material, the reinforcements and the mastery level.

It is essential for learning disabled students to be able to hold the directions for using a particular CAI package. An important selection criterion should be the simple operation of the program with frequent reinforcement of the instructions. The student must be able to interact with the computer without supervision for optimal use of a drill and practice program.

BEHAVIORAL OBSERVATION. An imporelement of the assessment of a learning disabled student is the observation of his behavior in the regular classroom. The consideration of how the student interacts within a classroom is essential in that the use of CAI will change those relationships. A student who is easily distracted within the classroom setting may be a good candidate for instructional programs because such instruction restructures the learning environment. Keogh (1971) reported that social and situational variables were important in increasing or decreasing hyperactivity. CAI typically takes place in an environment where there are few other students with whom to interact.

Unfortunately, even those packages which continually redirect the student's attention back to the program are easily ignored by a distractible student. The teacher should look for courseware which monitors for nonsensical or random responses and interrupts such a response pattern.

Some learning disabled students will attend to a task but will not analyze each alternative before responding. An appropriate program will force consideration of all alternatives before allowing a response, perhaps with a timeout penalty for impulsive responding. An ideal program will guide the student through each factor in considering the correctness of a response.

LEVEL OF ABSTRACTION. The test results should be reviewed to determine the level to which the student can abstract. If most of the student's responses to test items were very concrete, he is likely to get little benefit from most of the simulations on the market today.

Care should be taken in assuring that the student has the ability to generalize the lessons he learns on the computer to those which require oral, written, or performance skills. The teacher should follow up on each of the CAI lessons with a test of the skill in a different format.

SUMMARY. The learning disabled student's evaluation may be an effective guideline to selecting courseware. The student's teacher has more information concerning that student's needs than most teachers of nonhandicapped students. Unfortunately, whether the courseware is found is subject to the availability of adequately documented materials rather than the skills of the teacher in matching materials to students.

REFERENCES CITED

Jamison, D., Suppes, P., & Wells, S. The effectiveness of alternative instructional media: a survey, Review of Educational Research, 1974, 44, 1-67.

Jostad, K. & Kosel, M. Survey of commercial software, AEDS Monitor, 1981, 19(4) 19-21.

Keogh, B. Hyperactivity and learning disorders: Review and speculation, Exceptional Children. 1971, 38, 101-109.

Lindsey, M. State-of-the-Art Report: Computers and the Handicapped. Portland, OR: Northwest Regional Educational Laboratory, 1981.

Macleod, I. & Overheu, D. Computer aided assessment and development of basic skills. The Exceptional Child, 1977, 24, 18-35.

Modisett, D. Effects of computer-assisted instruction on achievement in remedial secondary mathematical computation. (Doctoral Dissertation, Fordham University, 1980). Dissertation Abstracts International, 994-A. (University Microfilms No. 8010982)

Reger, R., Schroeder, W., & Uschold, K. Special Education: Children With Learning Problems. New York: Oxford University Press, 1968.

Wallace, G. & Larsen, S. Educational Assessment of Learning Problems: Testing for Teaching. Boston: Allyn and Bacon, 1978.

"A MICROCOMPUTER-BASED INSTRUCTIONAL MANAGEMENT SYSTEM"

May, 1982

Lee J. Rodel
Evaluation Center
Hopkins Public Schools
1001 State Highway 7
Hopkins, MN 55343

ABSTRACT: A proven tool to assist the management of instruction is described. Used in an objectives-based curriculum, this tool provides information to teacher and student alike for use in instructional decision-making and results in improved learning. The tool is CAM - Comprehensive Achievement Monitoring. CAM operates on an Apple II microcomputer thereby making it accessible to virtually all educators. A demonstration of CAM on the Apple is included in the presentation of this paper.

I. INTRODUCTION

The ever-increasing demands for accountability in education have brought about a corresponding increased need for tools to facilitate the monitoring, management, and reporting of student learning. Such tools must have application at varied levels of instruction from primary through higher education, must be useable in curricular areas ranging from elementary school reading through graduate courses in biology, and must be available for use at the classroom or school building site. Such an instructional management tool must report data on student learning progress to students, teachers, administrators, and the public. This data is used by each consumer for the purpose of improvement of the learning process.

This paper presents an instructional management tool, namely CAM - Comprehensive Achievement Monitoring, which has a proven history of facilitating the monitoring and management of instruction thereby facilitating student learning.

Developed initially in the late 1960's at Stanford University, CAM has been in operation in the Hopkins, Minnesota, elementary and secondary public schools for over ten years. Considerable interest and a large number of adoptions of the CAM program have occurred during this time. Not until the last 1-2 years, however, has widespread adoption of CAM become a possibility. This is due to the fact that a version of CAM has been designed, developed, and is now operational on an Apple II microcomputer system. Now, with relatively minimal cost and effort, a proven instructional management tool, CAM, is available at the classroom and/or school level.

II. THE CAM MODEL

Research and development of CAM was begun at Stanford University in 1967 and continued at the University of Massachusetts by Dr. William Gorth and Dr. Paul Pinsky under a grant from the Charles F. Kettering Foundation to Dwight W. Allen, Dean of the School of Education. The techniques which developed partially out of this project to improve classroom evaluation techniques are known as Comprehensive Achievement Monitoring (CAM). Since that time, a large number of people have become variously involved with the techniques as researchers, implementers, technicians, proponents, and writers. Today, CAM provides a well-known model and a mechanism for integrating testing and evaluation into the instructional activities of the schools in a generally useful and more humanistic fashion.

CAM is a criterion-referenced evaluation system whose primary focus is on objective criteria defining the desired level of performance for pupils and programs. The CAM system's central purpose is to provide clear, relevant, and timely data to educational decision-makers at all levels of a school system. The types of data of primary interest in the CAM model are measures of the performance of the instructional programs of an educational system and of those treated by the system. The general purpose of the model is to provide dynamic means for maximizing the outcomes of instructional programs and the educational progress of individual students. Data on the performance of programs and students are collected and reported back to decision-makers in a format designed to support their decision-making activities. The decisions in turn direct the actual performance of the educational system to a closer approximation of expected or ideal performance.

The CAM model is designed to fit into the ongoing activities of a school district and help it achieve its own goals. No claim is made by the model for the correctness of any particular program content or organization. The CAM model is essentially a process through which the local educational system moves to obtain the information it considers relevant to its decision-makers. Though it is itself content-free, the process entailed by its implementation aids the district in defining and articulating its own goals and objectives. Instructional objectives must be specified and test items testing those objectives must be developed. Tests are then designed using those test items and administered in a planned evaluation program.

Appendix A contains sample objectives, test items, and pages of actual tests from a variety of curriculum areas/levels in which CAM is currently operational.

(For a detailed discussion of the theory and use of the CAM model, one is referred to a book entitled "Comprehensive Achievement Monitoring - A Criterion-Referenced Evaluation System" by Gorth, O"Reilly, and Pinsky, 1975, Educational Technology Publications.

III. CAM on the APPLE II MICRO-COMPUTER

The data processing support needed to utilize the CAM system was initially received by the Hopkins Schools through its educational computing cooperative using large-scale computers. Though this support continues even today for much of the daily CAM processing at Hopkins, it became clear that many schools around the country wishing to implement CAM encountered major problems in attempting to gain access to a similar large-scale computer support facility. The Hopkins Public Schools Evaluation Center staff therefore designed and developed a version of CAM which operates on an Apple II microcomputer system. This version of CAM, known to us "CAM on the APPLE," now has made possible the successful adoption of the CAM model in over fifty sites nationally ranging from Wrangell, Alaska, to Candia, New Hampshire, to Waycross, Georgia.

Stated simply and briefly, CAM on the Apple II stores instructional objectives and test answer key data, scores student test responses, and prints reports summarizing individual student and group performance to date on instructional objectives. CAM on the Apple II operates within either group-paced or individualized student progress instructional environments.

CAM operates on a minimum 48K Apple II dual disk drive microcomputer system with monitor/T.V., and hard copy printer. An optional card reader may be used to input student response forms.

Each CAM course has limitations as follows:

Course parameters:
- 100 objectives maximum
- 10 test periods maximum
- 20 test forms maximum
- 50 questions per test form maximum

Student parameters:
- 200 students maximum (group-paced)
- 100 students maximum (individualized)
- 4 teachers/student groups maximum

The CAM software on the Apple II is menu-driven. The user interacts at the keyboard with the software program to enter for each CAM course: (a) instructional objectives numbers, (b) test form answer keys, and (c) students in the course. Prior to each CAM testing period, instruction completion data is entered for each student or student group so that reports after test-scoring can reflect pre, post, and retention testing data.

Following initial entry of objectives, test form answer keys, students and appropriate instruction completion information, student test responses may be entered either at the Apple II keyboard or via optional card reader. Tests are then scored and selected reports printed in the desired report format. (See Appendices B and C for sample student response cards/sheets and sample CAM reports respectively.)

One of the most extensive and successful adoptions of CAM on the Apple II has been in the Yakima, Washington, Public Schools. According to Mr. Dan Penhallogen, Testing Coordinator, CAM on the Apple II was used in the math classes in eight elementary and two junior high schools and served twenty-four teachers and over six hundred students during the 1980-81 school year. Currently, CAM on the Apple II is servicing over 1200 students in forty-two classrooms in the Yakima Schools. An evaluation study was conducted in the spring of 1981 to determine the impact of the CAM system use on student performance. Analysis of study data indicated that the performance of CAM students was superior to that of their non-CAM counterparts. This statistically significant superior performance of CAM students indicates the effectiveness of the CAM system and replicates CAM evaluation studies conducted at other CAM adoption sites.

IV. WHERE DO WE GO FROM HERE?

Currently, versions of CAM on the Apple exist for use in both individualized and group-paced instructional settings. As a result of their use, enhancements have been suggested by the users and many of them implemented. At the time of the writing of this paper, a major project is underway which will result in the availability of an expanded instructional management tool for the Apple II. This enhanced tool will include all the capabilities of the current CAM on the Apple II, but will also include:

(a) expanded course parameters (number of objectives, students, etc.),
(b) objective mastery recording and reporting capabilities,
(c) handling of either individualized or group-paced
(d) increased ease of use ("user friendliness"),
(e) increased flexibility for test-scoring and report-printing,
(f) and other minor enhancements.

More information about the use and availability of CAM on the Apple II microcomputer can be obtained by contacting:

Hopkins School District 270
Evaluation Center/Room 246
1001 State Highway 7
Hopkins, MN 55343
Tel: 612-933-9230

226

WHAT IS REAL COURSEWARE?

Robert R. Rohm

Courseware Development Specialist, Scott, Foresman and Co.

Why define courseware? Or, even more fundamentally, why do we need the special term courseware? We are really searching for standards for a special genre of computer software: that which is created for educational purposes.

We who are interested in realizing the full potential for computer applications in education can assume that we are striving toward a certain high level of quality in educational software. Furthermore, we hope this level of quality will make the programs that comprise this category called courseware a substantial and important part of a student's education; it will make the programs much, much more than mere novelties or even supplements to his or her learning. We are looking for a way of defining and creating programs that can play a vital part in the education of children and adults. Courseware is the kind of educational programming that can have such a role.

I am suggesting four basic qualities that make educational programming courseware. First, it meets at least minimum standards for good software--that is, standards of computer program design and implementation. Second, programs are complete and integrated presentations of concepts or skills. Third, programs include useful and creative supplementary print materials. Fourth, they provide ways in which students can generalize learning they do on the computer to other media and to the world.

We expect all software we use to be useable and attractive. We expect clear instructions and content; readable text on the screen; a clean, effective method of getting the input the program needs, and of disposing with irrelevant or extraneous input; and a pleasing presentation, using where possible non-distracting graphics and sound. These standards for computer software should especially apply to courseware, for in order to learn, the student must find the medium comfortable, accessible, and non-frustrating. Courseware must be considerate of students and teachers.

In defining courseware we look for completeness and integration of content. The program need not be part of a series, but it should treat its subject thoroughly. Whatever the unit of instruction--whether it be a single subskill or a comprehensive treatment of a subject area such as reading or mathematics--it should have a beginning, a middle, and an end. It should make its purposes clear, and it should present interesting and appropriate content. Reading programs should feature literary variety; mathematics programs a variety of examples.

Finally, the program should give the student a feeling that he or she has completed the activity. The program can warmly (but not condescendingly) tell the student how he or she has done, or it can suggest ways in which the student can carry his or her learning over into other areas: "As you read, think about the problems story characters have, and how they solve them."

In essence, the content of courseware should be no less interesting and well presented than that of a good textbook.

Series and individual programs should be correlated to skills actually taught at specific grade levels. There should be a rhyme and reason to the selection of skills.

There should be relationship and interplay among the skills within each unit or module. This applies both to the selection of the skills and to their presentation.

Skills that comprise a unit can be tied together, and can illuminate or enrich each other. For example, characters, setting, and point of view are naturally interrelated as story elements, and their inclusion together in a single program module elucidates this interrelationship. On the other hand, there is no implicit relationship among the comprehension skills of recognizing problem and solution, cause and effect, and feelings of characters in stories, but putting them together can show how they might well affect each other in a story. A character's feelings can cause him or her to act in a way that creates or solves a problem. The computer offers an exciting potential for exploring such interrelationships. The student can select different combinations of characters and solutions and test them with the help of the computer.

Each module should be interesting in its content, graphic presentation, and in the variety of activities presented. The program should approach each skill in more than one way. For example, the program can offer the student the chance to study, practice, or apply the skill.

Within each lesson pacing, movement, interactivity, and content must be developed carefully. Paging should be user-paced; but more important than the fact that the child presses the enter key to go on is what happens after the child presses the enter key. Are concepts built gradually, with clear examples, and a chance to review? Are questions appropriately spaced throughout a lesson? What happens when the user presses a key or enters a word? It is true that the computer is infinitely patient; nevertheless, we can project onto the computer and feel that our continued wrong answers are an annoyance unless the program helps us out of our bind.

To be sure, the standards for content, skill distribution and coverage, and pacing apply to all instructional materials, but applying them to microcomputer courseware presents new and unique problems and challenges.

The test of courseware might simply be that learning is taking place. If it is courseware, then it should be much more than simply something to run on the computer; it should help students learn a skill or concept, or a body of skills or concepts. The courseware should include print materials that reinforce students' learning and that help students apply their learning through other media. Courseware should include supplementary worksheets, record-keeping sheets, and readings or activities related to the content treated in the computer programs.

Ultimately, software is courseware if it turns the computer into a means to access learning. Software, by converse, is something that allows you access to the computer; but to simply sit at the computer cannot be an end in itself. Courseware can help turn the computer station into an exciting place because it's a place where learning goes on. If the courseware students are using is of high quality, then they will be most conscious of the learning they are doing, and least conscious of the fact that they are doing it on a computer. That is the exciting part of it.

Toward a Functional Integration of Computers in the Educational Process: A Microcomputer Application for Facilitating Memorization

Kathryn J. Schilmoeller, Dennis R. King, Gary L. Schilmoeller
University of Maine at Orono

INTRODUCTION

Successful integration of microcomputers in the educational process requires the expertise of professionals in both education and computer science. The educator must assure that a well-planned systematic curriculum is in place with appropriate instructional goals and clearly stated behavioral objectives. If microcomputers are identified as part of the most relevant instructional strategies based upon students' needs and skill levels, the knowledge and skills of a computer specialist are desired in order to select adequate hardware, to locate or author software, and to provide technical assistance to the educator. Hopefully, these two professionals will work in tandem throughout an academic year to support the use of microcomputers in the classroom. The success and continuance of this special arrangement is measured by one ultimate criterion: Did the students learn?

Realistically, the exchange of information between the two professionals often is sequential rather than interactional. That is, an educator may express an interest in software to teach students how to perform a specific academic task, such as writing poetry. Ideally, the computer specialist identifies an available software program or writes an original piece of software, and the educator implements the program. In this instance, both would participate in the evaluation process.

Many times the process is even less interactive. An educator may want to utilize microcomputers in a classroom but have insufficient resources to commission a computer programmer. This educator must survey existing software available commercially or in published journal articles. In this context, the evaluation process that determines whether integration of microcomputer in the educational process was successful may break down. If the educator operates as the sole decision maker, several evaluation strategies are available to assist in the final decision on software. First (and best, if available) the firm offering software for sale or the published programs might provide evaluation data verifying that the software has been field tested with students. In the absence of such empirical data, the educator can evaluate existing software according to some conceptual framework. Does the software conform to existing data on learning theory; will the program present materials according to the overall instructional plan; will both teacher and student be able to execute successfully the mechanical functions that are required to use the software and hardware?

Ultimately, however, even if one or both of these evaluations are executed, the educator must rely on the empirical evidence from the educator's specific target population. The question to be answered is, do my students learn and do they learn more, faster, with fewer resources or whatever other criteria influence decisions pertinent to overall educational strategies. Regrettably, such empirical tests often are neglected by both professions. The following example describes an attempt to conduct an empirical test of the inclusion of a microcomputer program in the learning process.

EMPIRICAL EVALUATION OF "LEARNIT"

The three authors of this paper are all educators with varying degrees of experience with computers, but all are intrigued by the potential benefits to students created by actively including the computer in the learning process. Further, our committment is to the inclusion of the computer in the learning experience as a positive and fruitful experience both to the student and the educator.

Recently, the first author found a software program called "Learnit" published in a computer journal (Carlson, 1981). The instructions of the "Learnit" program best describe how it functioned:

> Press RETURN to see passage. When it appears, read it aloud very carefully. Then press RETURN again, and some of the letters will disappear. Read the passage again, and then press RETURN to eliminate some more letters. In ten passes all the letters will be gone. By then you will have it all memorized. (Carlson, 1981, p. 74).

If, in fact, this program worked as well as the instructions imply, the urge to implement it would be strong since memorization plays an integral role in the educational process. Foreign language learning, formulae, basic facts of introductory courses, memorization of poetry--the list of applications would be extensive. Supplemental information in the article tantalized us further. Conceptually, the program was founded on sound principles of discrimination learning (e.g., Bijou, 1968: Sidman & Stoddard 1966). And the last sentence in the article was very seductive--"When you try the program be careful to choose a passage you like--it could stick with you for a long time."

Nevertheless, not all of the learning literature uniformly supports the principle of gradual stimulus change as a functional teaching strategy (e.g., Schilmoeller & Etzel, 1979; Schilmoeller, Schilmoeller, Etzel, & LeBlanc, 1979). A search for empirical data to allay our concerns led to a dead end. No field test data existed. Consequently, we undertook the evaluation.

Our first step was getting "Learnit" to function on the computer. The member of our team who was the most naive in terms of the computer world thought we simply had to accurately type the program listing from the article onto the computer following the additional instructions provided by the editor for adaptation from a North Star Horizon to the TRS-80 Model I. Confronted with failure and after a series of frustrating debugging attempts. we decided to try the instructions for the Apple II, even though we were using a TRS-80. Imagine our surprise and delight when we found that this unobvious decision produced positive empirical results.

Finally the program was working--that is the computer performed the various functions that the article had described. Of importance in this initial step was the fact that the author most familiar with the computer was critical in assisting the other two in even getting set up to test "Learnit" as an instructional aid.

With the software and hardware functioning smoothly, we decided to test the program to the limits of its capacity. We choose a prose passage (from John Steinbeck's Grapes of Wrath) rather than poetry where rhyming or rhythm might facilitate memorization. The passage selected matched very closely the number of characters per line and lines per screen display allowed by the program on the TRS-80. Finally, since we wanted to assure ourselves that "Learnit" was going to facilitate memorization rather than just the novelty of repeated presentation of the full text on the CRT, we used a control group who saw the same passage ten times but without the random deletion procedure. In other words, this second group saw 10 displays of the entire passage without any changes.

Forty undergraduate students from an introductory course in child development were assigned randomly to the two groups. One group experienced "Learnit" and the other the full-repeat control procedure. To control a source of variance, the instructions for both groups were identical and did not describe the letter deletion procedure as had the instructions in the original manuscript. In accordance with the learning literature, the random deletion procedure alone was expected to be the critical variable.

The results dampened our hopes for "Learnit" but supported our suspicions concerning the effectiveness of the letter deletion procedure. First, "Learnit" failed to facilitate memorization. Second, subjects that experienced the full-repeat control procedure remembered a significantly larger number of correct words from the text. Third, neither group remembered more than 70% of the correct text.

CONCLUSIONS

1. The initial enthusiasm for "Learnit" has been dampened considerably. However, in going through the evaluation process, we have come up with a list of ideas for program modification that might make it more successful. For example, perhaps the original instructions were critical in alerting the students as to what they would be experiencing in going through the "Learnit" letter deletion procedure. Other potential modifications might involve how the letters were deleted.

At this point, we do know that "Learnit" as tested was not functional and that any modifications will have to be similarly tested for their functionality.

2. Our empirical test has given support to the need for educators to be cautious in the selection of software. Conceptual soundness does not mean automatically that the programs will facilitate learning.

3. The team approach was very profitable. Collectively, we had experiences in learning theory, computers, and evaluation research. Individually, any one of us might not have reached this point.

4. Our message is not to abandon the use of computers in the educational process. Rather, if one can find colleagues, what we presented here represents a formative evaluation of a program. Continued revisions of the software might lead to successful outcomes.

5. Finally, the ultimate criterion for determining if the integration of microcomputers into the educational process has been successful is whether or not the student learns. If he or she does not, the burden is back on us--both computer scientists and educators--to improve our strategies for collaboration in the development of instructional materials. Keeping this responsibility in mind we need to involve ourselves actively in the empirical evaluation of software. Such evaluations are critical to the continued successful integration of computers in education.

REFERENCES

Bijou, S. W. Studies in the experimental development of left-right concepts in retarded children using fading techniques. In N. R. Ellis (Ed.) International Review of Research in Mental Retardation. (Vol. 3) New York: Academic Press, 1968.

Carlson, Neil R. Learnit: A memorization aid program. onComputing, 1981, 3, 72-78.

Schilmoeller, K. J. & Etzel, B. C. An experimental analysis of criterion- and noncriterion-related cues in "errorless" stimulus control procedures. In B. C. Etzel, J. M. LeBlanc, & D. M. Baer (Eds.) New Developments in Behavorial Research: Theory, method, and application. *In Honor of Sidney W. Bijou.* Hillsdale, N.J.: Lawrence Erlbaum Associates, 1977.

Schilmoeller, G. L., Schilmoeller, K. J., Etzel, B. C., & LeBlanc, J. M. Conditional discrimination after errorless and trial-and-error training. *Journal of the Experimental Analysis of Behavior*, 1979, *31*, 405-420.

Sidman, M. & Stoddard, L. T. Programming perception and learning for retarded children. In N. R. Ellis (Ed.) *International Review of Research in Mental Retardation.* (Vol. 2). New York: Academic Press, 1966.

Steinbeck, John. *The Grapes of Wrath.* New York: The Viking Press, 1939.

"Change—Like It or not!"

by Lt Col (Dr.) Bill Wallisch

United States Air Force Academy

We who have gathered together under Orlando's generous sun are change agents of the first degree, often taking a societal heat for it that can match Orlando's in intensity. You know what I mean. We're the people who are trying to destroy the sacred places of learning with terrible machines and glowing boxes. Change agents or not, too often we're just plain not welcome.

There's an electronic revolution going on around us. We're being wired together while overhead more and more communications satellites are orbiting the globe. All of education is being caught up in this, as is every other facet of our society. As a matter of fact, we're much like Alice when she walked through the looking glass into a strange new world where objects were often unfamiliar. Educators look for blackboards and overhead projectors in the new world, but instead they are confronted with computers, videodisc, cathode ray displays, hand-held devices that speak several languages, and television sets the size of a wall. And the question is: What does it mean?

To those of us who feel most at home in this strange new world it simply means a new era in education, one that offers so much in terms of learning and ease of administration. Not since the invention of moveable type have educators had so much potential at their disposal in terms of tools for learning. Yet there is so much—coming so quickly—that many of our colleagues find themselves in a situation that is often very difficult to cope with. Innovators want to innovate, but with what? Adminstrators want to deliver systems. But which ones? There are so many devices, made by so many different manufacturers. There are so many claims made about the miracle of computer-assisted instruction, along with promises that rote-memory exercises will be completely handled by machines. And there are so many in the teaching profession who find all of that a threat. Like Alice, they're bewildered—and even threatened--by the strange machines and sights they find in this new world.

But we're sold. We're the people who sing the praises of this new technology, my gosh even let the machines live with us in our homes! We're confident of the promise all of the new technology offers, and point with pride at the things it's already done to make things better, easier.

But so often that same enthusiasm is not shared by our colleagues, or--worse--those up the chain who make the critical decisions in such matters as CAI. That's why I'm convinced that people like us must understand our crucial role as change agents and be ever conscious of our joint responsibility to help "gently and diplomatically" usher in this new educational age we know to be just around the corner. How well we perform as good-will ambassadors, however, might make a difference in how smoothly things will go.

I recently completed a two-year stint as a member of a state task force appointed by the governor to look into the implications of the new technology for higher education in Colorado. That task force encountered all of the things I'm suggesting about both the acceptance of and resistance to the new technology as it moved toward its final report.

Basically my own experience was that I found three distinct groups: groundswellers, neutrals, and resisters. As I made my way through my duties as a task force member, I soon learned that I had to approach each in a very different way. Those three types aren't limited to the state of Colorado, so I think you may have already encountered them in your area.

The groundswellers are neat. They're the faculty members who are writing their own programs and using techniques like CAI at the grass roots level. These are the English teachers who have written programs that drill basic sentence skills and even help students with rhetorical invention. They're physics, chemistry, and engineering professors who have introduced simulation and graphic displays of all sorts to their students. And they're the language teachers who can cut corners galore in French, Spanish, and German instruction. The list goes on and on. Problem is that they are doing great things at their level, with not much support from the top. Their work may not even be known to each other, because there is no means established to organize or combine their efforts.

Thus we meet the resisters. And too often these resisters are those who control the budgets and make the "go/no go" decisions about the use of computer related pedagogy in a school district or college. Once I was demonstrating an advanced handheld language tutor to a group. It was an advanced experimental model, given to me for special preview showings. And it really did put on quite a show. You could have heard that proverbial pin drop when this little baby started to rattle off sentences and give me vocabulary drill. It's eerie little voice was the only thing you could hear, even though we had a seminar room full of people. Now, don't get me wrong, most of the group was delighted with what they saw and heard, but one highly placed university administrator refused to even touch the little learning device when it came around to him. "I'm not going to hold that damn thing!" he said. And there, my friends, is the biggest problem (stumbling block) we face.

That outspoken stand AGAINST THE "DAMN THING" divided the group up, drew battle lines so to speak. The groundswellers protested, saying that the unit was wonderful. The resisters pointed at it and told them it would soon take jobs away from them all. And the neutrals? Well, a lot of them were impressed by the promise of the little gadget, but then a lot of them also knew that their collective personal futures depended upon the outspoken administrator who had rejected the teaching machine. It's not a good idea to speak up against someone who writes your ticket. Not in a meeting like that.

My role that day was that of change agent. But my mistake was that I demonstrated the gee whiz gadget before a group that was far too diverse. I had deans, full professors, and then a large number of lower ranking academics, including non-tenures. I put the deans on the spot, and I made them play a heavier hand than they might have had I gotten them by themselves. I would have converted more neutrals had I gotten them by themselves. And groundswellers? Well, they're sold anyway. I put them in a bad position at that, because I pitted them against big folks and made them take a stand unnecessarily. I should have saved my big gun support for a better occasion.

The real job of the technical change agent should be to bring all three factions together and have them working out the best plan for bringing about the new learning in their particular district, college, university, or state system. My initial approach to making that happen was the "Change—Like It or Not!" approach. For the most part it isn't sound (even though it happens to be true!).

After I analyzed the situation, I came to some conclusions about the three groups I was working with. For the most part the decision makers were worried about the potential disasterous impact the new technology might have upon their system, possibly even spelling massive teacher cuts or the actual demise of an institution or two. Coupled with that was a general lack of familiarity with things like CAI and little or no actual "hands-on" experience with any computer systems other than perhaps a pocket calculator.

The neutrals felt that same kind of threat, not to mention the kind of fear that comes with not keeping up with systems one sees peers and students alike involved with. Right now there are microcomputers humming away in the dorms, while some faculty dare to use nothing more complicated than an overhead projector in their classrooms. It can make some faculty feel very outdated and even isolated.

And the groundswellers? Us? Well, we don't need an introduction to the systems, but very often we're the ones doing the threatening. We don't mean to, but we can make the uninitiated feel very lost when we talk in glowing terms about our latest program for making students learn more.

So, with that in mind, I isolated those groups from each other and concentrated my efforts on each one at a time.

When you get the decision makers themselves you first acknowledge the fact that a new technology of any kind can bring with it the threat of potential harm. But leaders like themselves are in the positions they're in to wisely govern the system, make the right decisions, and set the kind of directions that will get the most out of the technology and protect the "patient." And the first order of business is to learn something of the new systems so that wise decisions can be made as they function in that systems delivery mode.

I suggest a hands-on session for top executives only, using the best diplomats you can find from the groundswell group. And as you're demonstrating the great advantages of CAI and how basics can be taught in such a convenient way, stress the point that the new machines and techniques will never replace the one-on-one wonder of master teaching.

You have the same kind of sessions for the neutral group, and it sure doesn't hurt to throw in some student demonstrators. It hurts a little, but the teacher must be reminded that he or she faces a generation of learners who more and more demand the new technology as a learning tool. Good faculty will get the bug as soon as it overcomes the initial fear of something mysterious and new. They'll soon be writing programs right along with the best of us.

And then get all of the groundswellers together—from all over your state or district—so they may exchange ideas, information, and programs. This will save duplication of effort and advance the cause of the new learning. Synergy is still one of the best words around. Talk to the innovators and organize them. They are the change agents who can do so much to pass new methods on to students, colleagues, and even those at the top. But—again—it has to be done with a concern for resistance to change. Sharing software among schools can save time and money. Once each level begins to exchange ideas (and fears), the future for CAI, instructional TV, and newer approaches to learning becomes brighter. But easy does it is the name of the game.

The message the change agent carries is the one everyone really wants to hear. Computers offer more effective use of faculty time, not to mention the promise of greater student learning gains. Under the direction of faculty, learners can be put on two-way interactive units that allow learners to make their way through material at a pace that suits the needs of each.

But education is not software. The microprocessor does not replace either the teacher or the classroom. It becomes the tool that allows the teacher the time he or she needs to carry out all important one-on-one teaching and at last be freed

236

to do more research and professional reading.

The American genius has made all of this possible. As a matter of fact, I don't believe that our strength comes from our coal supply or our synthetic fuel potential. I believe that America's greatest natural resource at this time is its leadership in microelectronic technology. We are decades ahead of our closest competitor. This microelectronic technology will put us at the forefront of world society, and make more and more of these marvelous tools available for education. And I don't believe that American industry wants to replace American higher education, though I know there are many in education who have that fear. If anything, I see a partnership between education and industry that promises a new road for each.

But I think higher education must assert itself, unify, and use the new technology fully. I think it must establish itself and its leadership in the new partnership. Gifted educators have always set the standards for education in this country. These are the people in the classrooms and the laboratories who touch students and bring a sense of value to lifeless test tubes, dictionaries, electronic circuits, and scalpels. Without these PEOPLE, and their sensitive approach to values, then the teaching machines will be useless.

To put it in the computer scientist's language, the teacher is the input the machine must have in order to have a useful message. But these teachers must not fear or reject the tools at hand. Rather they must take control of them. Just as blackboards stand cold and lifeless in the night, so will the new teaching machines be useless without gifted, sensitive, human leadership.

Change is certain. And those of us who see its implications must do the job of bringing the best of it to our American educational system. We are more than computer oriented educators for we also bear the responsibility of the change agent.

I'm reminded again of Alice's trip through the looking glass. Just as she entered Wonderland, Alice encountered the Cheshire Cat. He appeared above her in various stages. Finally when he was whole, little Alice asked him which way she should go. To that the Cheshire Cat replied, "Where did you want to go?" Alice told him she really didn't have any idea. And that made the Cheshire Cat smile his broad smile and say, "Then it doesn't make any difference which way you go, does it?"

If our educational system is to go in the right direction, it's up to us take the lead. Change is coming, and it's our job to make it meaningful. We're the change agents.

UNDERSTANDING THE "CAI PHENOMENON"

George W. Bright
Department of Mathematical Sciences
Northern Illinois University
DeKalb, Illinois 60115

Computer assisted instruction (CAI) has repeatedly been shown to be at least as effective as, and more efficient then, other instructional techniques. This phenomenon can be explained in terms of the time-on-task hypothesis. This explanation also suggests ways of making future improvements in CAI software.

Computer assisted instruction (CAI) has been studied for about two decades. One of the most exciting results reported has been that CAI oftens generates learning undifferentiable from that generated by other techniques, but CAI seems to accomplish its goal in significantly less time. This result will be referred to as the "CAI phenomenon." Without an understanding of instruction this result appears to be, and in fact was initially interpreted as, an enigma. Current understanding of instruction, however, suggests both a possible explanation for the CAI phenomenon and a framework within which further improvement of computer assisted instruction can occur.

The CAI phenomenon is important for a variety of reasons. From a teacher's perspective, efficiency in instruction even for limited topics would enable either more time to be devoted to complex topics or more topics to be taught. In either case, learning would be improved. For administrators, efficiency can improve productivity or cut costs for fixed amounts of learning. For curriculum designers understanding the CAI phenomenon might suggest ways to improve other types of curriculum products, again improving productivity or cutting costs. For researchers, such understanding may yield a better grasp on how people learn.

The discussion of the CAI phenomenon will be divided into four parts. First, an overview of the research literature will establish that the CAI phenomenon exists. Second, the Beginning Teacher Evaluation Study (BTES) will be described and hypotheses relevant to the CAI phenomenon will be presented. Third, these hypotheses will be used to build an explanation for the CAI phenomenon. Fourth, implications of this explanation will be drawn for both practical and theoretical development of CAI.

Research Support of the CAI Phenomenon

Computer assisted instruction materials typically have an advantage over traditional materials in that the sequencing of concepts and the branching decisions based on students' responses are quite thoroughly thought through. Hence, it should not be a surprise when CAI materials are equally as effective as, or more effective than, traditional instruction. However, when these effects are observed in significantly less time, or at a noticeably faster rate, then there is cause for careful analysis of results.

For this paper documentation of the CAI phenomenon will be provided by analyzing reviews of research on CAI, rather than by searching original sources. Molnar (1973) in citing a 1970 paper by Bunderson stated that "CAI permits students to achieve educational objectives in much less time and that saving of up to 40% is not uncommon (p. 64)." Cody (1973) agreed that "most of the studies... indicate that material can be taught by computers in substantially less time than with conventional techniques, with no loss in achievement (p. 25)." Neither review indicated the applicability of this conclusion either by grade level or by content.

Jamison, Suppes, and Wells (1974) were more precise though somewhat less enthusiastic, in stating that "at the secondary school and college levels... CAI is about as effective as traditional instructional when it is used as a replacement. It may also result in substantial savings of student time in some cases (p. 55)." Edwards, Norton, Taylor, Weiss, and Dusseldorp (1975) noted that in nine CAI studies in which time was measured all reported that CAI was faster with equal or superior achievement in each case. Seven of the studies involved college-aged or adult subjects, and six dealt with science-oriented content.

More recently Thomas (1979) reviewed CAI studies in secondary school and concluded "that CAI reduces the time required for a student to complete a unit (p. 109)." The content of studies he reviewed ranged across mathematics, accounting, industrial arts, biology, and typewriting. Overton (1981) in a review of CAI studies in mathematics noted that in two post-1975 studies CAI produced results faster than regular instruction. In most studies she reviewed, however, time differences were not reported.

The BTES

The major focus of the Beginning Teacher Evaluation Study (BTES) was to identify and describe teaching skills and their impact on student outcomes (Powell, 1980). One of the most important variables identified by the BTES associated with learning was academic learning time (ALT). For any academic task, there is an upper limit to the school time available for this task (allocated time), and a student will be attentive to the task during only part of this time (engaged time). The success which the student encounters during this engaged time provides the means for defining ALT. High success describes situations in which the student makes only occasional careless errors; low success describes situations in which correct responses are made only at about the chance level. ALT is the engaged time spent working with high success (Fisher, Berliner, Filby, Marliave, Cohen, & Dishaw, 1980).

For the purposes of relating the BTES to the CAI phenomenon, several of the conclusions are important (Fisher et al., 1980).

1. The ratio of engaged time to allocated time is positively associated with learning.

2. The ratio of ALT to engaged time is positively associated with learning.

3. The ratio of low sources, engaged time to total engaged time is negatively associated with learning.

4. Accuracy in diagnosing student skill levels is positively associated with achievement and ALT.

5. Prescription of appropriate tasks is positively related to student achievement and success rate.

6. More substantive interaction (e.g., presentation of information, feedback) between the student and an instructor is associated with higher levels of student engagement.

The common thread of these conclusions is the mediating effect of ALT on achievement. The so-called "time on task" hypothesis can be summarized as "The more ALT a student accumulates, the more the student is learning (Fisher et al., 1980, p. 8)."

Synthesis

The BTES time-on-task hypothesis provides one possible explanation of the CAI phenomenon. If the hypothesis is true, then it is reasonable that a given level of achievement should on the average be produced by a fixed number of units of ALT. Thus, if the engaged time with a constant ratio of high success can be increased, or if the ratio of high success within a fixed amount of engaged time can be increased, then there should be greater learning. Looked at from the perspective of achievement at a predetermined level, as the engaged time increases or as the ratio of high success increases, the total clock time needed for learning decreases. This is precisely the CAI phenomenon.

The amount of time students are willing to spend with CAI materials is noticeably longer than with most other instructional techniques. Teachers who have begun using microcomputers typically note almost immediately that students have to be pushed out of the computer lab at the end of the day. Newsmagazines (e.g., Time, All Things Considered on National Public Radio) have also documented this phenomenon. Similarly, commercial video games are so popular that city councils are trying to restrict access by public school students. Perhaps this increased involvement is related to the substantive interaction between the user and the machine, as suggested by the BTES findings. All of these observations point to the apparent conclusion that CAI increases engaged time, at least in absolute terms.

CAI also seems to increase the ratio of high success within engaged time. In very sophisticated CAI programs, the questions/exercises/problems presented are matched to student's achievement on previous, similar exercises. Low success items are eliminated from consideration. For less sophisticated programs, great care typically has gone into the program design so that the question/exercises/problems are chosen with the likelihood of high success. In both cases, high success rates are built into the programs.

Thus, the creation of CAI programs has often been completed so as to mazimize ALT. This has in the past been accomplished without the theoretical construct of ALT, so the maximization has not been explicitly intentional. However, the qualities that characterize the best CAI have had this effect, and this effect alone could account for the efficiency that has been experimentally observed.

Speculation

It would seem, then, that the efficiency of computer instructional software can be increased as the proportion of ALT within a program is increased, either by increasing the amount of engaged time at high success or by increasing the rate of high success within the engaged time. The amount of engaged time might be increased through increasing the interest of the student; i.e., in some general sense, the motivation of the student. Graphics, varied pacing, and fantasy (e.g., Malone, 1980) are three possible ways of doing this, though many more might be revealed by a search of the literature on motivation. Increasing the success rate would seem to come mainly from matching the student's past performance to the questions/exercises/problems presented. Much more needs to be known about diagnosis, however, before this can be done with great skill.

Designers of CAI software should keep in mind that their decisions will affect ALT. Recognition of this may create an awareness of appropriate learner-machine interaction that will greatly ease the development of CAI programs. Too, within the ALT context bits and pieces of instructional design; e.g., state objectives, outline the program in successively greater detail; seem to fall together and to be important for one basic reason; that they all impinge on ALT. Interpretation of the CAI phenomenon within the ALT context also puts the design of CAI into the "whole cloth" of effective instruction rather than just into the more narrow, albeit exciting, world of CAI. The unity provided by this interpretation is important for progress toward the creation of a theory of CAI design.

References

Cody, R. Computers in education: A review. Journal of College Science Teaching, 1973, 6, 22-28.

Edwards, J., Norton, S., Taylor, S., Weiss, M., & Dusseldorp, R. How effective is CAI - A review of the research. Educational Leadership, 1975, 33, 147-153.

Fisher, C.W., Berliner, D.C., Filby, N.N., Marliave, R., Cahen, L.S., & Dishaw, M.M. Teaching behaviors, academic learning time, and student achievement: An overview. In C.Denham & A. Lieberman (Eds.), Time to learn. Washington, DC: National Institute of Education, 1980.

Jamison, D., Suppes, P., & Wells, S. The effectiveness of alternative instructional media: A survey. Review of Educational Research, 1974, 44, 1-67.

Malone, T.W., What makes things fun to learn? - A study of intrinsically motivating computer games. (Report CIS-7, SSL-80-11) Palo Alto, CA: Xerox Research Center, August 1980.

Molnar, A.R. Computer innovations in education. Washington, DC: National Science Foundation, February 1973. (ERIC Document Reproduction Service No. ED 077 731)

Overton, V. Research in instructional computing and mathematics education. Viewpoints in Teaching and Learning, 1981, 57(2), 23-36.

Powell, M. The Beginning Teacher Evaluation Study: A brief history of a major research project. In C. Denham & A. Lieberman (Eds.), Time to learn. Washington, DC: National Institute of Education, 1980.

Thomas, D.B. The effectiveness of computer assisted instruction in secondary schools. AEDS Journal, 1979, 12, 103-116.

MICROCOMPUTER APPLICATIONS IN THE COMMUNITY COLLEGE

Dr. Donald A. Dellow
Dr. Lawrence H. Poole
North Country Community College
20 Winona Avenue
Saranac Lake, NY 12983

In the past year, it would have taken a total recluse to have avoided the media coverage of the impact the microcomputer was having on society. What Toffler (1980) predicted about the coming revolution seemed to be occurring more quickly than he or anybody else expected. In almost every news magazine and professional journal, articles have begun to appear detailing the numerous uses for which the microcomputer is suited. By far, the most prolific uses for the microcomputer seem to have been documented in the area of business. Educators seem to have adopted the microcomputer as the long awaited workhorse to realize earlier dreams of computerized instruction. Most educational institutions might take some pointers from their business colleagues and use the microcomputer for the multitudinous purposes to which it is suited. North Country Community College is trying to do just that.

In January of 1981 North Country Community College established a microcomputer laboratory with the goal of providing hands-on experience for business and secretarial science students. Seventeen microcomputers (15 CMBs and 2 Apples) and several thousand dollars worth of business software were purchased. Students were given just a brief overview of programming in BASIC and were then immersed in wordprocessing and countless other business applications. The philosophy at the college was one of preparing people to use microcomputers for real applications, those already understood and those yet to be discovered. It was felt that the quickest way to an employer's heart was to provide students who were not afraid of the computers and were capable of using them for wordprocessing, complex problem solving, database management, and bookkeeping.

In stressing the practical uses of microcomputers, it became obvious to us at the college that we needed to demonstrate how we as educators and administrators were using them. We thus embarked on a journey of "technological application." We found that every use to which microcomputers had been put in industry and business was applicable to our situation. The rest of this paper documents some of the ways in which we have been using microcomputers.

ADMINISTRATIVE USES

The list of uses for microcomputers in the administrative area will grow as we use our imagination. We have found, however, that three general types of programs have been able to solve almost any type of problem with which we are presented. Commercially available software packages for wordprocessing, electronic worksheet problem solving (i.e., VisiCalc) and data base management provide the means for creatively addressing many administrative questions. Administrators with restricted budgets and expansive desires to use microcomputers would do well to consider the purchase of general utility programs rather than programs written and sold for one restricted purpose. A little creativity and some good programs can save a lot of money.

One of the first major applications we found for microcomputers was in the area of wordprocessing. The advantages of being able to compose, edit and print final copies of grants and correspondence was immediately obvious. For those who can't stand messy scribbling and scratch-outs in their writing, the wordprocessor is their salvation. It is very rewarding to see one's writing efforts come out of aprinter looking neat and organized. Rewriting is not the chore of cut-and-paste and having the secretary type just one more rough draft. The primary author feels the wordprocessing has improved his writing considerably, he now takes time to edit-as he always knew he should!

One of the business packages purchased for the college microcomputer lab was VisiCalc; the software package that has been reputed to have been the motivating force for thousands of executives to purchase a microcomputer. VisiCalc is an electronic worksheet with 60 columns and 254 rows. The first use found for this program was in budget preparation and maintenance. With VisiCalc it is possible to design what might be called a template or model. Tentative accounts and budget codes may be established in a columnar table and formulae built in to have the program do all of the subtotals and totals. When estimated expenditures exceed anticipated revenues, individual items need to be modified. With VisiCalc, or another similar program, one simply changes line items to new amounts and the program instantaneously recalculates all totals. This allows a "what-if" kind of playing with budget figures that is not practical when using a calculator and paper and pencil. In addition, the chance of making a simple calculation error is minimized when one does not have to refigure everything.

Currently at North Country Community College, the Dean of Academic Affairs and two of the three Division Chairpersons use VisiCalc to design, maintain and check budgets. In years such as this one, where six budget revisions were undertaken, the amount of time saved by using a microcomputer is considered a true investment.

In a similar vein, the Dean of Academic Affairs was called upon to do a cost analysis study of the various academic programs in the college. A microcomputer and a program entitled Desktop Plan, from the same company that developed VisiCalc, was used to do a detailed cost analysis study of the college. In the time it would have taken to explain to a programmer what was desired, the complete study was designed, completed and printed for the timely review of the college board.

Another significant use to which the microcomputers have been put is in the area of database management. So far the microcomputers have been used for mailing lists, inventories of equipment, car registration, student sticker identification and listings of the several hundreds of microcomputer programs available at the college.

One last major administrative use we have found for the microcomputers is in the area of electronic gradebook maintenance. Approximately twenty-five percent of the faculty are now maintaining a copy of their gradebooks on VisiCalc. The ease with which formulae can be built into a file makes the tedious task of calculating averages, subtotals and summations a thing of the past. By establishing a file on VisiCalc, a teacher can simply update the gradebook periodically and give students current information on the total point accumulations and percentages. With a flick of the risk a printed copy is available for posting or backup. Students are anxious to have the feedback and instructors have better data on which to advise and counsel individual students. The automatic recalculation functions of VisiCalc also reduce the possibility of calculation errors.

INSTRUCTIONAL USES

It seems that each week new instructional uses are found for the microcomputers. The initial instructional applications were concentrated in the business and secretarial science areas. An introductory course in data processing was the first course to require students to complete activities on the microcomputers. Students were given several projects that demonstrated the basics of programming, sorting, data base management, and computer assisted instruction. The emphasis was on computer literacy, as opposed to programming. Well over 100 students were introduced to computers in this manner.

The secretarial science instructors began to use the microcomputers to introduce word processing. In less than ten hours of lab time, the students developed a high degree of competency on a word processing package that would allow for the latest state-of-art editing, text storage and retrieval, form letter production and numerous other text-editing functions.

In another secretarial science course the students were introduced to the calculator mode of the microcomputers and simple programs that provided loan amortization schedules, interest calculations, and other general business applications. Again, approximately ten hours of hands-on instruction were provided.

In response to student interest for more knowledge of the applications of microcomputers, a course entitled "Microcomputer Literacy" was developed and offered in the summer. Several faculty, some students, and a number of townspeople took the three-credit hour course that emphasized hands-on experience and knowledge of microcomputer applications.

As a result of the strong applications orientation of the microcomputer lab on campus, additional faculty began to explore possible applications in their subject matter areas. The administration of the college supported the efforts to introduce the micros in other areas by providing an in-service workshop and allowing faculty to use instructional budget monies to purchase software packages. Consequently, barely eight months after microcomputers made their debut on campus, instructors in ALL of the major instructional programs were beginning to work microcomputers into their courses. After one year, seventeen of thirty-four faculty members on campus were using the microcomputer for some instructional purpose.

In the humanities, a music instructor purchased the necessary software packages and peripherals to teach music theory on an Apple II. Additional funds are being sought to establish some computerized electronic pianos on which music theory and music instruction may be combined.

The English and communications faculty are now having their students exposed to word processing by requiring at least one paper to be completed on the microcomputers using the word processing software. The initial reactions this semester indicate the students will end up writing all or most of their papers on the computers.

A reading specialist is assisting students by having vocabulary review programs available on microcomputers for those courses that are notorious for their jargon. Through this process, students with reading problems are able to have the drill-and-practice necessary to build their vocabularies outside the normal testing activities of the course.

In the sciences, microcomputers are now being used in biology, chemistry and environmental conservation. In each case, available software programs allow for demonstrations or simulations of complex processes not normally available to our students. In the mathematics area, all three instructors are beginning to provide students with review exercises on the microcomputers. Additional instructors from the social sciences and radiological technology have also made inroads in utilizing the microcomputers.

The need for computer literacy has been so widely accepted on our campus that a proposal has been submitted to make it a part of graduation requirements, beginning in the Fall of 1982. The plan calls for students to demonstrate both a competency in terminology and hands-on skills through the completion of coursework or individual study on modules and computer-assisted instructional materials. Whether students are going directly into the job market or are transferring to a four-year college, there is agreement that computers will be a part of their lives.

STRATEGY FOR THE FUTURE: MORE MICROCOMPUTERS

At NCCC a decision has been made to invest in more microcomputers for the instructional and administrative functions of the college. The one minicomputer on campus will, for the foreseeable future, be relegated to college-wide administrative functions and microcomputers will address all other needs. The microcomputers have been found to be better adapted to our needs than the minicomputer. Besides, the great number of persons, both staff and students, who have been exposed to computing are partial to their personal step into the information revolution!

REFERENCE:

Toffler, A. The Third Wave, New York: William Morrow & Co., Inc., 1980.

PROCEDURES FOR DEVELOPING ADMINISTRATIVE SOFTWARE

By Duane Loewen
MECC

Developing software for Administrative purposes requires several procedures:

1. ANALYSIS
2. RESEARCH
3. DESIGN
4. DEVELOP and EDIT
5. PROGRAM and GRAPHICS
6. DOCUMENTATION

The first step in designing a computer application for Administrative applications is the selection of the topic and formulating the learning objectives.

There are several systematic approaches to Administrative design.

STATE OBJECTIVES

After the topic has been selected, the specific objectives to be achieved should be listed. Listing the objectives will assist in achieving logical sequential steps in the instruction. When objectives are defined, the designer must decided which parts of the task the computer will perform. The method used to prepare the objectives should be that which is supported by the participant's district.

HOW WILL IT FIT INTO YOUR PLAN

The layout and design of an administrative computing application may very well be the most important factor to producing a quality product. It must be specific as well as appropriate. Develop a statement how the materials will be incorporated into the administrative setting.

APPROPRIATION OF MICROCOMPUTER

The designer should select an idea that is appropriate for the microcomputer.

SELECT A STRATEGY

The designer must select the type of strategy to use. This strategy will partially depend on the objective to be achieved and how the program will fit into the administrative program.

THE ROLE OF SUPPORT MATERIAL

When deciding how the computer material will be incorporated into the program, the role of support material and other media should be considered. How much of the support material will be presented on the computer or will they be presented in the support material.

STAGES OF DEVELOPMENT ANALYSIS

First a survey of the user community must be made to see if there is a need for the proposal. Secondly is it a good fit for a computer. Thirdly discuss with State Department and administrative specialists. Fourth is it important. Then present the plan to a management team. A MECC staff member or ISD developer receives an idea for a new product from a school district in the form of a program description with specifications. A unit outline should be included if an entire diskette is planned. There is a discussion as to MECC's interest in the concept and how MECC could assist the person in developing a product.

RESEARCH

First analyze the administration concept being proposed. Secondly determine the role of: computer, peripherals, documentation and other media. Thirdly present an outline to the specialist and state department.

Fourthly is this a reasonable approach and is the project feasible. Then develop timeline and cost to the design manager.

DESIGN

First develop objectives for each concept. Secondly develop flow of the program and select a sample of the resulting program. Do screen layout for each concept and present this to the design manager. If this is acceptable the programmer's code is developed and a timeline is developed. If the design is not acceptable the developer in charge re-writes the design specifications as necessary before any programming is done. Support materials are begun here too and their development will be in parallel with the program.

DEVELOP AND EDIT

The developer along with the senior programmer will design the program before any programming is done. Top-down design, structure design where applicable, and structured walkthroughs where applicable will be used to complete this phase. The program is coded, implemented, tested and revised by the programmer assigned to the project.

Questions are asked such as: Is the order good? Does it flow? Is it a good use? Is the program ready for special features?

PROGRAM AND GRAPHICS

Programmers take the screen layouts and codes and incorporate artistic design and final programming techniques such as design timing, error checking and provide details. After the walkthrough committee of developer and programmers has approved a product then the developer will begin a more formal review and testing of the program.

DOCUMENT

Support materials are being worked on in parallel with the program. Design outline for documentation and a rough draft is turned over to the developer for review. Documentation for an individual computer program might be thought of as a 'book within a book'. It has a beginning called the introductory page; a middle section with background information and administrative use; and an end with handout pages and sample runs. After the middle section of the support book are written; that is, after each computer program is written-up to be a self-contained unit, the back pages of the support book are drafted. The back pages are for reference. They contain information which may be essential for some readers to refer to and can be cheerfully ignored by others.

If there are a number of back pages of distinct content, grouping them as Appendixes A, B, C, etc. has two advantages: First, the appendixes signal a clear separation from the main body of text. Secondly, 'appendix' is a useful word to head a miscellanea of information in a table of contents. Whether or not they appear as appendixes, back pages of distinct content aways bear their own title such as 'Glossary of Terms', 'Technical Instruction', 'Sources', etc.

The front pages of the support book are standard: outside cover with title, author, and/or organization - inside title page - copyright notice when applicable - table of contents - introduction (may be preceded by a preface and lead into a lengthy overview) - acknowledgments. In addition to the standard front pages, an index to the computer programs, with a descriptive caption for each program, gives readers a bird's-eye view of what is expected.

TEST

Field testing or administrative testing of the program and support materials. The developer reviews the review and others may be called on to assist. If further revision of the program and support materials is necessary they will be completed.

Bullet proof testing by programmers that did not program for this product. Code is revised as necessary. Editor reviews the support booklet and makes the necessary changes. When this has been all finalized the package is distributed.

THE USE OF GEOPROCESSING IN EDUCATIONAL RESEARCH AND DECISION SUPPORT

Porter Sexton
Management Information Specialist
Portland Public Schools
Portland, OR 97227

ABSTRACT:

Geoprocessing systems employing a Geographic Base File and address matching techniques are now available to many school districts and educational institutions. These systems are most often used for bus routing and scheduling and for redistricting or determining boundary changes. The Portland (Oregon) Public School District geocodes a variety of data, including student performance data, census and demographic data, taxation data, and election data. Standard statistical procedures are then used to produce a wide variety of analyses useful for decision support. One of the most promising developments to come from this method of analysis is the ability to retrieve student data from selected geographic areas. This paper presents a discussion of this analysis technique and some of the analyses which have been done.

GEOPROCESSING:

Geoprocessing is the linkage of geographic analysis and data processing.

Two particular functions of geoprocessing in widespread use in educational system planning are Dual Independent Map Encoding and address matching.

The Dual Independent Map Encoding (DIME) technology was first developed by the New Haven Census Study in 1967 and has been substantially enhanced by developments by the U.S. Bureau of the Census and a number of agencies working in the area of geoprocessing. Essentially, DIME processing codes the nodes (endpoints) and line segments which form map polygons (geographic boundaries, e.g., city blocks) and independently codes map polygon designators for each delineated area. Algorithms then analyze the coded data to ensure completeness and compatibility of the map elements.

Address matching entails the construction of computerized Geographic Base files which relate addresses to specific geographic indicators such as census tract and block or latitude and longitude. Computer programs can then match addresses against the Geographic Base files and create records which contain the geographic indicators for the addresses. For public school districts, address matching can be particularly useful in determining such information as the school attendance area in which students reside.

GEOGRAPHIC-BASED INFORMATION SYSTEM:

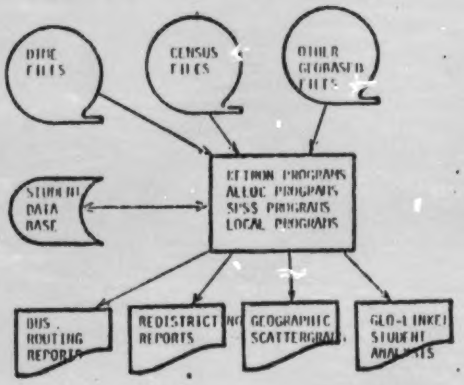
Two major applications of geoprocessing by school districts have been in the areas of bus routing and school system redistricting--the redrawing of school attendance boundaries. Over the past two years, the Portland (Oregon) Public School District has been heavily involved in both of these activities.

In addition to these two standard applications of geoprocessing, Portland Public Schools has developed two rather innovative applications which have proved valuable in educational decision support. The genesis for these applications is found in Dr. Kenneth Dueker's "An Approach to Integrated Information Systems for Planning" [1], in which the basic possibilities and constraints of geoprocessing-based computer information systems are discussed.

At Portland Public Schools, a student data base contains records of individual student performance such as test scores, grades, attendance, suspension and/or expulsion; individual student demography, such as student address, ethnic code, and home language other than English; and student program, such as class enrollment, talented and gifted program, and special education services. The student's address is address matched to such geographic indicators as school attendance areas, census tract and block, and latitude and longitude in Oregon Standard Coordinates.

These geographic indicators are linked to further geographic-based data, such as census data, tax assessment data, and voter turnout and voter support data. Computer programs access, analyze, and report this interrelated data as needed. The following applications illustrate some of the uses of this geoprocessing capability for educational research and decision support.

Figure 1:
PORTLAND SCHOOLS GEOBASED INFOSYSTEM



BUS ROUTING:

In the area of school bus routing, the district uses DIME files originally obtained from Multnomah County (Oregon) and updated locally and software obtained from Ketron, Inc., in Cambridge, Massachusetts. [2] The resulting more efficient routing, coupled with staggered school starting times, saves the 52,000 student district approximately $1,000,000 per year in transportation from a total transportation budget of $7,000,000.

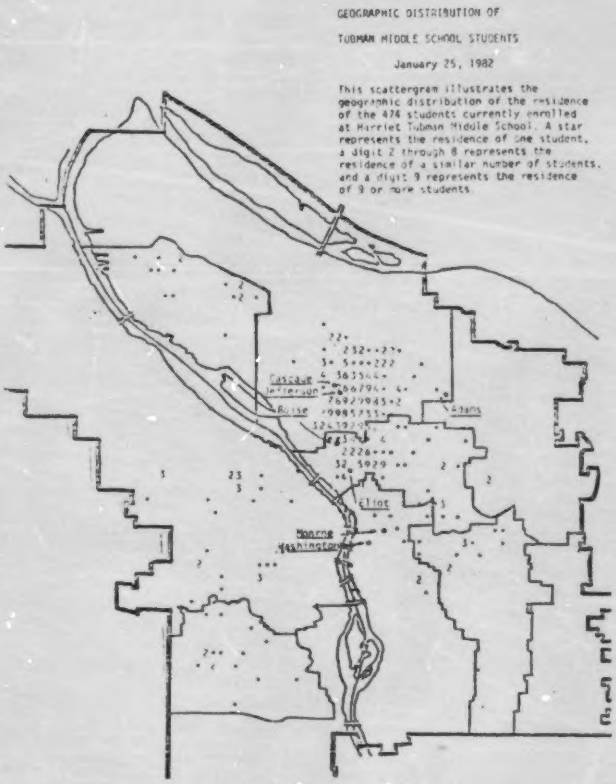
REDISTRICTING:

In the area of school redistricting, DIME files and Ketron software have also been used. Over the past two years, the district has been involved in the process of closing three high schools and four elementary schools due to enrollment declines. Computer programs with heuristic solutions to location-allocation problems were obtained from the Department of Geography at the University of Iowa [3] (Programs: ALLOC and LAP). These programs analyze the output data obtained from address matching student address records with geographic indicators, and identify the school sites whose closure would necessitate the smallest increase in total student travel distance to school. Once those sites are projected as closed, the most distance-efficient allocation of students to the remaining schools, within constraints such as capacity of school buildings, is calculated. The Portland Public Schools Board of Education voted to close the three high schools whose closure necessitates the smallest increase in total student travel distance to school, as identified by this geoprocessing methodology.

SCATTERGRAMS:

An innovative geoprocessing application developed at Portland Public Schools is the production and use of geographic scattergrams for decision support. Latitude and longitude figures for each student's address are attached to each student's record by the address match system discussed above. Specific student records of interest can then be selected by the value of any specific data item or items on the student's record. These selected records are then analyzed using the SCATTERGRAM subprogram of the Statistical Package for the Social Sciences (SPSS) [4] software available from the University of Kansas. Parameters in the SCATTERGRAM subprogram allow the user to adjust the scale of the output graph to match the scale of existing maps, either printed or on data files. The output graph can then be overlayed on the map--manually in the case of printed maps and automatically (by computer program) in the case of data-coded (digitized) maps. An example is shown here of a geographic scattergram thus produced to aid the Board of Education in determining the site for the location of a school. Similar geographic scattergrams have been used at Portland Public Schools in the determination of program sites and in the reassignment of students with conversions from elementary schools (K-8 grade configurations) to primary and middle schools (K-5 and 6-8 grade configurations).

Figure 2:
GEOGRAPHIC SCATTERGRAM



GEOGRAPHIC DISTRIBUTION OF
TUBMAN MIDDLE SCHOOL STUDENTS
January 25, 1982

This scattergram illustrates the geographic distribution of the residence of the 474 students currently enrolled at Harriet Tubman Middle School. A star represents the residence of one student, a digit 2 through 8 represents the residence of a similar number of students, and a digit 9 represents the residence of 9 or more students.

STUDENT ANALYSIS:

Another application of geoprocessing developed at Portland Public Schools has been very simple but very effective in research efforts. This application is the grouping of students by geographic indicators. Students who reside in a given block or set of blocks can be easily identified. With the recent acquisition of 1980 Census Summary Tape File 1 [5] for the Portland metropolitan area, block level data containing socioeconomic and educational level data is available. This data becomes linked to individual student records as discussed above. Thus, groups of students with similar socioeconomic home environments can be identified.

Such groups of students can then be studied in terms of the multitude of individualized student data available on the district's computer files such as test scores, grades, attendance, and behavior as demonstrated by the incidence of suspensions and/or expulsions.

An illustration of this application may be of interest. Students from a particular geographic area of Portland were found to have a higher high school "dropout rate" than students from any other section of town. Their dropout rate was uniformly high at their neighborhood high school and at all other high schools in the district which they attended except one. Their dropout rate from that one high school was relatively low.

The neighborhood high school for the area of town with the high dropout rate was closed this year. Many students residing in this area were then assigned to the high school where the dropout rate had been lower. (This student assignment was not done in an overt attempt to reduce dropout rates.) Although it is too early to tell conclusively, preliminary indications are that dropout rates for students from this area are somewhat lower this year.

CONCLUSION:

The total costs for the acquisition, operation, and maintenance of these systems, including personnel costs, have been about $150,000 per year, far less than the $1,000,000 annual savings on transportation. Not only have these geoprocessing systems thus been extremely cost-effective, they have greatly enhanced our analytical and decision support capabilities as well.

The use of geoprocessing in such standard applications as bus routing and redistricting, and such new applications as geographic scattergrams and grouping of students for analysis based on geographic factors has been an increasingly valuable activity at Portland Public Schools over the past two years.

REFERENCES:

[1] Dueker, Kenneth, "An Approach to Integrated Information Systems for Planning," Computers in Local Government: Urban and Regional Plannning, Auerbach, 1980.

[2] Kindler Associates, Inc., Computer Aided School and Transportation Assignment, Ketron, Incorporated, 1979.

[3] Hillsman, Edward L., Heuristic Solutions to Location-Allocation Problems, University of Iowa, July, 1980.

[4] Nie, Norman H., et al., Statistical Package for the Social Sciences, McGraw-Hill Co., New York, 1975.

[5] Data User Services Division, Summary Tape File 1, 1980 U.S. Census of Population and Housing, U.S. Department of Commerce, Bureau of the Census, Washington, D.C.

COST EFFECTIVE COMPUTER GRAPHICS TERMINAL UTILIZATION: THE STUDENT AS TEACHER

L.V. Brillhart/Eric A. Bell
Triton College

ABSTRACT

The combined pressures of increasing enrollments, decreasing budgets and rapid advances in computer technology, place educational institutions which wish to keep up-to-date in a virtually untenable position. Large class sizes and limited budgets do not allow for purchase of sufficient equipment for all students. Even if sufficient equipment could be purchased, rapid technological advances, would render the equipment obsolete. Colleges must, therefore, devise educational strategies to optimize use of limited equipment.

Triton College, with the aid of two NSF grants has devised a system of allowing 100 students per semester to be introduced to computer graphics with the use of only two terminals, one digitizer, one plotter and one printer. Each student must spend at least one hour per week on the computer. An additional forty-five hours per week are available to all students on a first come first served basis.

INTRODUCTION

Courses in graphics have become an increasingly smaller component in the engineering curriculum. Twenty years ago, most schools required a course in descriptive geometry and one or even two courses in drafting. This has decreased to a single three semester hour course combining descriptive geometry and drafting required in only some curricula. The advent of CAD/CAM has revived an interest in graphics, specifically computer graphics. Since the engineering curriculum is already overloaded, computer graphics has frequently become a component of the traditional graphics course.

The incorporation of computers into a graphics course was by no means an easy task. Computer terminals or micro/mini computers were expensive, as was peripheral equipment. Industrial utilization progressed rapidly yet very unevenly. No "standard" equipment existed. Much of the equipment used by industry was way beyond the scope of even the wealthiest school. Apart from cost, other severe problems existed. Class sizes were large and teaching loads heavy. Graphics instructors rarely were well versed in computer utilization or programming. The course content in a graphics course was already extensive and little time was available to add more material.

A number of systems were considered by engineering schools. Some schools purchased "turn key" systems already used by industry[1]. Others used batch processing with large main frame computers[2]. Some schools offered computer graphics as an elective course. The wealthiest schools set up individual stations for each student, and the poorest simply had to ignore the situation.

However, an even more critical issue than finance arose. What should be taught in computer graphics? Should students simply learn to access existing programs and use the computer as a drafting instrument? If so, what programs and equipment would best prepare them for an engineering career? Should students learn to program a computer graphics terminal? Should the computer graphics equipment be used to enhance three-dimensional visualization?

Triton College, in developing a computer graphics component for the existing engineering graphics course decided to follow a philosophy consistent with that employed by all engineering education. It is the mission of engineering colleges to teach fundamental concepts and practices that would be useful for the forty-year career of a practicing engineer. The young engineer in serving her apprenticeship would acquire skills and procedures for a particular industry, company or department. Computer graphics emphasized the skills needed in controlling a medium so as to achieve visual results needed. The logical thinking processes required

253

were paramount, with equipment operation and program utilization important only as it contributed to the main focus. Cost reduction procedures and time constraints were also dealt with only when methods could be developed which enhanced skills needed to successfully use CAD/CAM.

COURSE DESCRIPTION

Engineering Graphics as taught at Triton College is a 3 credit hour course meeting 6 hours per week. The course combines descriptive geometry and graphics, teaching "on-the-board" and computer applications simultaneously. An abridged course outline is shown in Fig. (1).

Contents	Weeks
Graphs, Tables, Charts	1
Point and Lines	4
Lines and Planes	5
Isometric, Orthographic	3
Cabinet, Cavalier	1
Sections	2
Dimensioning, Tolerancing	2
	18

FIG. (1)

Except for the first week, the course is a traditional descriptive geometry course followed by a drafting course. The sequence allows for the drafting portion to be based on theories learned in descriptive geometry.

The course is taught in a lecture/lab format. Each meeting is started with ½ to 1 hour lecture followed with a lab experience. The lab consists of both on-the-board drawing and computer graphics. Some students work on the board while others work on the computer. Each student spends an average of one hour per week on the computer.

The classroom is laid out as a lecture drafting area with an adjacent computer lab. Students working in both areas can be assisted by one faculty member (no teaching assistants exist at Triton).

STUDENT AS TEACHER

Triton College with the financial help from two NSF grants (NSF/CAUSE and NSF/ISEP) was able to acquire two Tektronix 4052 computers, a digitizer tablet, a printer and a plotter. This equipment is allocated primarily to the graphics course. Since graphics class sizes varies between 24 and 28, the student/computer ratio is at least 12/1. The only feasible way for each student to receive a comprehensive experience on the computer is to cycle students through the computer while other students continue the paper and pencil portion of the assignment.

The procedure used in teaching the computer component of the graphics course consists of an initial introduction of the topic by the teacher followed by a sequential learning and teaching experience by each student. The teacher, during lecture, gives the assignment for the day or week, explaining the procedure to be used and expected output from the computer. After the lecture is complete and plates have been assigned, two students are randomly chosen to start the learning/teaching chain. The teacher shows both students how to use the the procedure for the day. This procedure can be the use of a canned program, learning how to operate a peripheral, learning BASIC or writing a program. Each student then becomes the teacher and shows the next student how to do the assignment. This procedure continues until each student has completed both portions of the cycle, student and teacher. The professor intervenes only when a major problem occurs. Because of the physical layout of the teaching area, the professor can monitor and help students on the board and on the computer. Professor intervention is obviously needed when equipment malfunctions or the paper supply is depleted. In the beginning of the semester, when computer anxiety is rampant, faculty intervention is also required when two particularly fearful students in a row are assigned to a terminal. This situation is remedied by the third week when the random choice of order is replaced by assigned order. The order is made so that, if possible, two computer nuts or two computer haters do not pair up. The professor also intervenes when timing goes out of schedule.

This method would rapidly deteriorate to a game of telephone if it relied solely on verbal transmission of information. To prevent this a programmed learning manual has been prepared for each computer station. The manual contains material shown in Fig. (2).

1. Graphing
    - Data Graphing
    - Pie Chart
    - Text Writer
2. Points & Lines
    - True length, slope, bearing
    - Keyboard Entry
    - Digitizer Entry
    - Shortest distance between 2 lines
    - Horizontal distance " " "
3. Lines & Plane
    - True size, dip & strike
    - Intersection of line & plane
4. BASIC Programming
    - Entry/Output
    - Arithmetic Operations
    - Graphics Operations
    - Loops
    - Arrays
    - Data Statements
    - Character Strings
    - Interaction with Peripherals
    - Interaction with Other Computers
5. Isometric/Orthographic
    - Isometric Rotation
    - Isometric-Orthographic
    - Drafting
6. Cabinet, Cavalier
    - Drafting
7. Sections
    - Drafting
    - Shading
8. Dimensioning/Tolerancing
    - Drafting

FIG. (2)

The manual also contains a listing of control characters and the operations they perform. The meanings of all error messages is also given as are directions for simple editing procedures.

The manual contains a guide to the use of canned programs as well as a rudimentary guide to the BASIC language. Modules used at the beginning of the semester are extremely detailed in instructions. An excerpt is given in Fig. (3).

IF YOU WANT TO DRAW A PIE CHART:

1. Turn on the terminal.
2. Insert tape DATAGRAPHING
3. Press AUTOLOAD
4. COMPUTER: Menu
   User: 2 $\boxed{R}$
   COMPUTER: To enter your own data enter Y
   User: Y $\boxed{R}$

FIG. (3)

With each sequential module, old material is removed and assumptions of competency are made. These decisions are reviewed each semester and changes based on student input made.

The teaching of BASIC is also initiated in a step-by-step manner. A sample program is given with each new concept documented. Directions for use of the program are then given. Comprehension is tested by asking each student to write a program using similar concepts. An example is given in Fig. (4).

TO ENTER NUMBERS AND TO PRINT LETTERS OR NUMBERS

```
  Program                            Documentation

100  INIT                        Clears Variables
110  PRINT "ENTER A"             Prints "Enter A"
120  INPUT A                     Accepts input and labels it "A"
130  PRINT "ENTER B"
140  INPUT B
150  C=A+B                       Performs arithmetic operation C=A+B
160  PRINT "C EQUALS ";C         Prints "C Equals" and numerical value of C
170  END                         Indicates to computer that program is over
```

INSTRUCTIONS

1. Turn terminal on

2. Wait until the cursor is at the top left corner. If the screen flares, press the HOMEPAGE key.

3. Type 100 (please make sure to use 0 for zero and O for letter "O") and press AUTONUMBER key.

4. Enter the program. The computer will automatically number the lines.

5. After you have hit the RETURN key after typing END, hit the autonumber key again. This stops the automatic numbering.

6. Type: RUN and hit the return key.

7. Enter values for A and B as requested.

TESTING OF COMPREHENSION

Write a program which finds the hypotenuse of a triangle given the length of the other two sides.

FIG. (4)

Again, as the learning of BASIC continues, documentation is decreased and instructions are minimized. As has been stated earlier, this material is being taught in a first semester course. Triton College has an open door policy, hence, while most engineering students are well prepared, some are deficient in reading and math skills. The manual is written so that students with pre-requisite skills can experience a modicum of success on the computer.

RESULTS

The effectiveness of the student-teacher procedure can be studied in two ways. All students must write an individual project at the end of the semester. The rate of successful completion of the project is one measure of the effectiveness of the pedajogical method. All students complete projects. The projects vary in difficulty according to student's background. Students who concurrently take chemistry or physics write quite interesting programs in those areas using computer graphics. Other students write graphs or math programs or even games.

While computer graphics was incorporated in the course to introduce students to a tool of engineering, it was also hoped that the use of computer graphics as an education tool would enhance students' three dimensional visualization skills. A standardized final exam was given in the engineering graphics course. The experimental group (using computer graphics terminals) received a mean grade of 81 on the final exam while the control group received a grade of 67. This is particularly significant because the experimental group had a mean math ACT score of only 14, while the control group's score was 20.

CONCLUSIONS

Students can be taught computer graphics with a minimal outlay for equipment and no additional faculty or personnel. This can be done by optimization of equipment utilization and use of students as teachers. The method of peer tutoring described does not rely on "brighter" students as tutors but places all students in the role of student and teacher. Students as teachers, while at times inept, do much to alleviate computer anxiety and tend to procede at a pace appropriate for a tyro.

CREATIVE PROGRAMMING FOR YOUNG MINDS

In the summer of 1979, Computer Related Enrichment and Technologically Inspired Viable Education (CREATIVE) was born. It's mother was necessity, it's father was hope. And it came into this world laughing, smiling and full of life.

It was created to provide young minds with an opportunity to improve their thinking skills, logic, and natural curiosity and creativity.

Those aspects of CREATIVE that did not inspire young minds to think or to develop good learning skills, were quickly cast aside, and replaced by something better.

Microcomputers were selected to be the media of communication, because it was believed that this generation would find themselves either masters of this new technology, or slaves to it, and we had no desire to create minds that could be enslaved.

It became obvious quite quickly, that there were three major impediments to their rapid learning. (1) The lack of printed materials they could understand, (2) the absence of materials that would allow real independent progress and branching, and (3) examples that were understandable and of interest to young minds.

We wanted to instill an enjoyment for learning and pride that comes from being able to reason things out on their own. We also wanted them to become independent learners, and to develop their own self-motivation. Creativity and problem-solving ability were also very important to us. We did not want to concentrate on solving problems that had already been solved many times before; we wanted to develop the ability to solve problems that did not yet have answers.

This sort of project required setting up a special environment. A structure was needed for support, but not for dependence. We needed ample opportunity for branching, without requiring any particular branch. We needed a reward system that would encourage creativity, independent thinking and logic oriented processes, but would not inhibit self-paced learning.

The challenges were there, the students were there, and the time was right.

Now, almost three years later, we find ourselves riding the crest of a wave of success. CREATIVE centers are springing up all across the nation. Teachers that we have trained are starting programs both within schools and outside of schools. An approach that was intended for gifted children has also proven successful with hearing impaired, EMH, average and remedial children. The materials we developed are being used to train both children and adults...often in the same room and at the same time!

Let's take a look at the CREATIVE approach, and see if we can see what makes it so successful.

Almost everyone remembers Dorothy in the Wizard of Oz. She was told that if she followed the YELLOW BRICK ROAD, she would arrive at the Palace of the Wizard. Once there, she assumed that all of her problems would be solved by the Wizard. What people often overlook, is that the wisdom that Dorothy gained during her travels, all came when she left the YELLOW BRICK ROAD! The road was always there for her to return to when she felt the need to move on, but the real learning experiences took place off the road, when she was experiencing situations that were real to her.

The road provided structure, it got her from place to place. But she decided when to use it. To provide such a road, we created a series of instructional manuals that were custom made for each major brand of microcomputer. The manuals were designed to be easy to read, without the language of the professional programmer or the style of the commercial publishers. Devin Brown, a professor of children's literature at Eastern Illinois University, was selected to write the original series, and did an excellent job of incorporating within the manuals, his own style of humor. The instructional series included seven such manuals, each using one of the seven colors of the rainbow, red, orange, yellow, green, blue, indigo and violet.

To provide an opportunity to use their new knowledge and to gain confidence in their thinking ability, projects were added to the end of each manual. These were colored to match the cover of the book. They were designed to require the student to understand what he had learned in the white pages of the manuals, and to apply, analyze and synthesize that knowledge so that it could be used to solve a problem not previously discussed. Confidence in the ability of the students to use their brain to think and perform logical operations resulted in this phase of CREATIVE becoming our SCARECROW.

Our TIN MAN sprang from the need to have a stout heart. Gaining a better understanding of one's self while struggling to get the "bugs" out of a program requires a stout heart. To be willing to push oneself to the brink of frustration... and then to back off, and hit it again, until finally it cracks beneath the logical, persistent attack of a creative mind requires perserverence.

The courage to put your ego on the line and tackle problems that face the real world is not something that students are often asked to do. Usually, only those problems for which the teacher already has answers are assigned. We wanted a set of real life problems for which answers were not provided. We called these our ALL STAR PROJECTS. A manual of these special challenges was created for use after each of the last five instructional manuals. It would require courage on the part of the student to even try these, and great persistence to solve them. These ALL STAR PROJECTS became the LION of our Oz.

Computer games became the sunshine of our Oz. They provided warmth and a cozy environment in which to rest and relax for a short while. They also provided surprising growth in creativity and technique. Students would look at the LISTing of these game programs to see how certain objectives were accomplished, and often gain new insight or ideas that would later be useful to them. We found that the process of creativity usually started off by copying...then altering... then putting together parts from a variety of sources... then finally developing an original idea or solution of their own.

Several methods of reward were developed. The solution to the last colored project in each manual is sent to CREATIVE CENTRAL in Charleston, Illinois. There a laminated colored card with the programmer's name and programming level is produced. This is then sent back to the student, showing his level of expertise. In addition, his name appears in the next months newsletter, CREATIVE Creations. The cover of the newsletter features the graphic work of a student each month. And many of the articles and programs are written by young programmers. The newsletter is designed to be a forum for such contributions.

The greatest reward, of course, is the satisfaction that comes from taking on a problem or programming challenge and solving it.

Finding gifted young people that can do wondrous things with a computer is not unusual. You read about them all the time and probably you know several yourself. But we were finding a room full of them. And some of them had come from remedial classrooms in the public schools!

These young minds were teaching a microcomputer how to solve problems with its rather limited BASIC vocabulary. And in the process, they were not only improving their own basic skills, but were developing problem-solving techniques that they could build upon as they continued through life.

It is not all just a field of roses however. Remember the poppy fields of the Wicked Witch that dull the senses and put one to sleep?

The boredom and closure of repetitious Computer Assisted Instruction is a Wicked Witch. Our Ruby Slippers are the natural curiosity and creativity of each active young mind.

And the Wizard of Oz? You remember that when Dorothy and her friends finally survived the hazardous trip and stood before the Wizard to ask their favors, they found they already had acquired that for which they were searching.

I find that I have become like the Wizard of Oz, watching the students during their journey through CREATIVE, gather unto themselves loyal friends such as logic and reasoning, stoutness and persistence of heart, and intellectual courage. They started out looking for an answer, but during their journey along the YELLOW BRICK ROAD they discovered for themselves how to find their own answers.

They are now ready to plan their own journey!

- Marilyn Buxton
Henry A. Taitt

MICROCOMPUTER PATIENT EDUCATION COURSEWARE

Kent Canine, PhD
Mae McWeeny, RN, BSNEd

Eighty percent of the United States population lives with low back pain that often could have been prevented with the knowledge and practice of proper moving and lifting techniques. Quite a challenge for health educators! Back injury prevention education programs are becoming more and more widespread--in industry, through health education, via employee wellness or lifestyle management programs and as part of therapy regimens for people with back discomfort or chronic low back pain. The development of these programs reflects the new dimension of using formalized patient education as a part of health care.

Abbott Northwestern Hospital/ Sister Kenny Institute provides low back education at all of these levels. Abbott Northwestern Hospital/Sister Kenny Institute is an 838 bed major medical center located in Minneapolis, Minnesota. The medical center is an acute facility as well as a rehabilitation center. In addition, Abbott Northwestern Hospital, Inc. opened an Occupational Medicine Clinic in 1981 to serve business and industry in preventing and managing work related injuries. The Institute for Low Back Care, located in the medical center campus refers patients for low back education and treatment.

Abbott Northwestern Hospital admits a substantial number of patients to the acute hospital with a primary diagnosis of low back pain. Ninety percent of the patients in the resident chronic pain rehabilitation program have chronic low back pain. In addition, patients treated through the Institute for Low Back Care are admitted to Abbott Northwestern Hospital.

Although the variety of our low back education is extensive, ongoing evaluation indicated a need for revision of the "low back school" format and content for inpatients. This evaluation raised three important questions. Was the content consistent? How individualized was the instruction? And most importantly, were our clients able to adapt their moving and lifting habits once they were back in their own environment?

In an attempt to address the continuity issue, a two-part slide/ tape program was developed in 1979. This became the primary teaching resource used in our low back school for the hospitalized patient with low back pain.

In our evaluations of the low back school, it became evident that we had indeed provided continuity from class to class and from teacher to teacher by the use of this audiovisual program. But the program didn't solve all of the problems. The class was still long, often patients nodded off or couldn't tolerate sitting for long periods, since sitting can be very painful for the patient. And we still did not have individualization of instruction or active participation by the patient.

A review of the literature was undertaken to seek a solution to these problems. We found that even though microcomputers have had limited application in the education of patients, potential for this use was vast. Therefore, it was decided that computer assisted instruction could provide a method that was consistent in content, individualized, interactive, and cost effective.

Our next step was to establish parameters for the development of the microcomputer courseware. These parameters fell into three primary categories: 1) physical status of the patient, 2) learning readiness and motivation of the patients, and 3) resources in our health care facility.

Physical status of the patient included recognition of factors such as activity tolerance, pain level, status in treatment regime, and progress toward resumption of normal activities of daily living.

261

Another parameter was the learning readiness and the motivational level of the patient. The patient hospitalized with low back pain recognizes a need to change and exhibits a willingness to modify his/her lifestyle. In most instances the patient wants more information about his/her ability to function, which suggests learning readiness.

A third parameter was the availability of resources in our health care facility. Abbott Northwestern Hospital/Sister Kenny Institute had several resources upon which to draw. These included staff expertise in the provision of patient education, and in the treatment of low back pain. There currently existed a variety of methodologies for the provision of low back education, all of which could be easily adapted to the development of microcomputer courseware. The low back injured patient population, as previously mentioned, is substantial in numbers and reflects patients at different levels in the continuum of care, from acute to chronic. Our health care facility provides the learning environment and administrative support for the development of innovative treatment in low back care and the development of educational methodologies consistent with this treatment regime.

In review of these resources, it became obvious that our organization did not have expertise in microcomputer courseware technology. For this reason a consulting relationship was established with a firm which had expertise in microcomputer courseware technology and educational methodology. Productive Learning, Inc., a nationally recognized courseware company, was selected. This company had extensive experience in development of courseware for the General Mills Corporation, McGraw Hill Publishing Company, Control Data Corporation and other organizations. Its president and primary consultant in the development of the low back education courseware is Dr. C. Barry Borman.

The development of the low back education courseware consisted of several stages. The first was review of existing course material. These materials included slides/tapes, audiovisuals and print. They were perceived as being suitable for adaptation to microcomputer courseware. The second phase of the program development consisted of staff training both for Abbott Northwestern/Sister Kenny Institute and Productive Learning, Inc. Productive Learning, Inc. provided a microcomputer awareness workshop and education to the hospital administrators and educators on the conversion of existing low back patient education programs to microcomputer patient education programs. In turn, the patient educators provided an understanding of the special needs of low back patients to the staff of Productive Learning, Inc.

A field test version of the low back courseware was developed for use on the Apple II Plus microcomputer. The courseware included computer managed instruction features such as provision for restarts, recording of student data, recording of student responses for assessment and evaluation purposes. The courseware was field tested on patients, public, health educators and subject matter experts. As expected, the field testing and evaluation suggested that certain modifications needed to be made in the courseware. These modifications included development of additional graphics, greater emphasis on need for behavioral change, and specific content changes. These modifications were incorporated in the final product.

A microcomputer system was installed in a location which was convenient and accessible to patients at all hours. This met the patient's need to complete lessons when he/she was not in therapy but yet not in pain. The Apple II systems are on a portable unit so the patient can use the key board with ease from his/her wheelchair or bedside. In addition, the patient may elect to work at a centrally located learning center.

This flexibility is essential in adapting the use of microcomputers for patient education.

The courseware is used as one of several teaching alternatives in our low back education program. It consists of two diskettes. The content of the courseware has been divided into four parts. Each portion is short to accomodate the patient's comfort level and attention span. Part one begins with a lifestyle assessment to help the patient identify risk factors which must be adapted to protect his/her back. A printout of this assessment is reviewed by the patient educator so low back school sessions can be individualized to the patients in attendance. Part two is on structure and function of the back. This part of the courseware can be completed at any time during the patient's hospital stay. It is designed to help the patient better understand the normal back and the rationale for certain aspects of medical treatment. Posture and body mechanics (part three) demonstrates how proper posture and patterns of movement influence daily living activities. The patient completes the fourth part on moving and lifting before coming to low back school. The basic rules for moving and lifting are illustrated to show their relationship to stress on the low back. These rules are then practiced in low back school.

This combination of patient involvement via the microcomputer plus low back school has placed emphasis on education as a viable aspect of treatment. The patient leaves the hospital having begun his/her lifestyle adaptations which, if practiced, will help protect his/her back from further injury.

The potential use of microcomputer courseware within our own hospital has just begun. The employee back education program, patients in the chronic pain rehabilitation program and outpatients treated through the Institute for Low Back Care have access to the low back diskettes. Educators are incorporating them into the total education program, often with a different emphasis depending on the population using the program.

Courseware related to the area of lifestyle management would have potential not only in the low back education program, but also with patient education programs for people living with other chronic conditions. We have identified an expanded curriculum to help change lifestyle behaviors such as living with stress, adjusting nutritional needs, energy saving helps and work simplication, and other adjustments to living with a chronic condition.

The low back courseware is being marketed through the Abbott Northwestern/Sister Kenny Institute's Publications Department and through Productive Learning, Incorporated, both in Minneapolis, Minnesota.

MICROCOMPUTERS IN COOPERATIVE GROUPS

BY

KEN AND MAGGIE CORENS

The initial wave of excitement that arrives with a new microcomputer ebbs quickly when the instructor thinks, "Oh no! Only one computer and thirty kids...What do I do now?!" Or are you one of the fortunate few with enough computers to fill a computer center? In either case, after a short time you start wondering how to use the micros more effectively.

How can we make all the students aware of the computer's possibilities? How can we even get them a chance to operate the new marvel? What can we do about the limited number of computers and the large number of eager, anxious students? COOPERATIVE GROUPS may be your answer. By grouping on one computer, a teacher is able to get a whole class through a particular lesson in two or three days rather than two or three weeks.

Let's say you decide to review nouns and verbs on the computer. Begin by telling the class they will be going to the computer in groups. Each group will have a color name and everyone in the group will have a different role. The roles are starter, listener, scribe, encourager, checker, and observer (more on this later). An easy way to assign roles is to put each role and its explanation on a color coded card, and then randomly distribute the cards to the students. With a class of thirty, you would have six groups of five--the red group, blue group and so on. Groups can be as small as three or as large as six people. Review the roles with the class, modeling the behaviors you want to practice. Make clear where the Scribe and Listener should turn in their reports. Make sure the class knows what the assignment is, and what written work you expect of them. It's also a good idea to have a particular goal for a group to work on each time. For example: "Today's goal is that each person have a turn at the keyboard."

Now you are ready to go. Pick a color group and say to them, "OK, red group, you can be first to the computer. Red starter -- here is the disk. Listener and Scribe, do you have your reports? Does everyone understand their roles?" Send the red group to the computer.

A group will be at the computer for ten to twenty minutes at a time. You may want them to complete a certain task (do the nouns and verbs lesson), or have each group stay at the computer a designated time (15 minutes). It will take two to three days to get a class through a nouns and verbs review using the computer one hour a day.

After all groups are done, evaluate with the class not only the subject matter they completed, but how they worked in their groups. Did they take turns, listen, agree, praise? How could they make their cooperative groups better next time?

Why the emphasis on cooperative groups rather than just groups? Recent research shows that people who fail in jobs often do so because of a lack of interactive skills, rather than a lack of technical skills. Computers provide a perfect opportunity to practice both of these skills. Appropriate interactive skills can be learned through the cooperative group roles mentioned earlier.

Here is an explanation of what each role involves:

STARTER

1-Starts things (turns on computer and monitor, puts in disk).

2-Reads beginning directions out loud.

3-Has first turn.

4-Calls on people for turns at the keyboard, makes sure everyone participates.

5-Puts things away if this is the last group to use computer.

SCRIBE

1-Has paper and pencil.

2-Keeps the written records for the group.

3-Asks each individual to sign the work when finished.

4-Is in charge of turning in written work.

Note: Entire group does the work together--the scribe is merely the recorder.

ENCOURAGER

1-Encourages group to support each other, be accepting, and make positive statements.

2-Contributes encouraging statements when possible.

LISTENER

1-Reminds students to listen, pay attention, not to interrupt.

2-Listens for the particular goal of the day, such as participation, discussion, reasoning, group decision-making, agreement, listening, or encouraging statements.

3-Fills out listener's report.

Sample Listener's Report:

"LISTENER'S REPORT"

sign in here ------

	🙂	😐	☹
took turns			
listened			
discussed			
praised			

Our group was good at (put answer on back)

CHECKER

1-Checks for discussion and agreement in group before answers are entered at the computer.

2-Contributes encouraging statements when possible.

Note: This is a very important step because many of the software programs do not offer the chance to change answers. What is entered gets stored on their record. Checking does not mean the group must reach consensus. The person at the keyboard asks whether everyone agrees. If they do not, the group then has the opportunity to teach its members, to disagree, and explain other ideas. One person cannot just disagree with an answer without good reason. There always seems to be a few students who realize the power of the veto and try to overuse it, but once the students learn the veto doesn't have power without persuasion, they quickly stop disagreeing.

OBSERVER

1-Quietly observes the group in operation.

2-Keeps notes on the process of the group.

3-Shares observations during follow-up of the lesson.

Note: Observing is a difficult role. This task is best used by older students familiar with the group skills. This role is most often played by the teacher.

If you have several computers--enough so the entire class can be grouped at one time--you are in for a special treat. Once the class has learned some of the basics of cooperative learning, you will find students very involved with their group, and that you have a luxury teachers only dream about. You will have the opportunity to step out of your direct teaching role, unencumbered by someone tugging at your elbow for help, and observe the environment, behaviors, and interactions in your class. When was the last time you had a chance to do that?

However, whether you have many computers or only one, setting up and using cooperative groups is an effective way to use the microcomputer. It's a good feeling to know you are sending students to the computer in a responsible, organized manner.

REFERENCES

Diane K. Browne (612-933-9205)
Cooperative Learning Projects
Hopkins School Dist. 270
125 Monroe Avenue South
Hopkins, MN 55343

Cooperative Learning
University of Minnesota
202 Pattee Hall
150 Pillsbury Drive S.E.
Minneapolis, MN 55455

Ken Corens (612-933-9230)
Instructional Computer Consultant
Hopkins and Eden Prairie Schools
1001 State Highway 7
Hopkins, MN 55343

Maggie Corens (612-473-9791)
Wayzata School Dist. 284
726 East Rice Street
Wayzata, MN 55391

J&J Book Educational Materials
162
Windsor Lane
New Brighton, MN 55112

MICROCOMPUTERS IN SPECIAL EDUCATION--BOON OR BANE?

Alonzo E. Hannaford
Associate Professor of
Special Education
Western Michigan University
Kalamazoo, MI 49008

Microcomputers can be valuable tools in the management and delivery of instruction to help meet the cognitive, affective, physical, sensory, and learning needs of the handicapped. Their use in assessment, prescription, formative evaluation, resource allocation, and report preparation can be a boon to special education instructional management. In delivery of instruction their ability to facilitate providing individualized, motivational, interactive instruction can also prove to be a boon. It is possible, however, for them to become a bane due to problems which exist with the educational compatibility, instructional design adequacy, and technical adequacy of much educational software. Inappropriate use of this technology with the handicapped may also serve to make it a bane to special education.

By definition handicapped learners have needs which differ either in nature or degree from those of the non-handicapped. To help meet these needs various forms of special education have evolved. Whichever form is used, however, the task remains the same; to help meet the needs of the handicapped learner.

Handicapped students represent a diverse array of learners who may demonstrate needs in one or in a combination of areas.

Cognitive Needs.

Some handicapped student have quantative and/or qualatative difficulties in thinking, remembering, creating, solving problems, synthesizing, evaluating, etc. Such difficulties necessitate education which is well analyzed and sequentially presented at the learner's own pace.

Affective Needs.

For learners with social and/or emotional difficulties special needs are found to be associated with their interests, likes, dislikes, attitudes, motivation, and interpersonal relationships.

Physical Needs.

When learners possess disabilities which reduce their neuro-muscular capacity, their needs in school center mainly around modifications or adaptations to their learning environment to permit them to function as near normal as possible.

Sensory Needs.

Learners with auditory or visual handicaps represent special problems educationally. Their needs tend to center around the use of methods, materials and/or devices to permit them to compensate for their sensory deficits or to supplement their remaining sensory abilities.

Learning Needs.

Some student possess the cognitive ability to learn and do not have problems in the affective, physical, or sensory areas but still have difficulty learning. They may have needs in comprehension, written and/or oral expression, reading, and/or mathematics. Frequently problems with perceptual distortions also generate needs in this area.

Anything which enables the process of education to be delivered more effectively and efficiently is looked upon as a boon. A technology that has the potential to do just that in special education is that surrounding microcomputers. This potential exists both in the management of instruction and in instructional delivery.

INSTRUCTIONAL MANAGEMENT

Legislative mandates surrounding education of the handicapped brought with them massive management requirements. Within the instructional management area microcomputers can be a very valuable ally to the special education teacher. More specifically microcomputers can be of assistance in the following areas:

1) Assessment--Prior to planning for instruction it is necessary that the handicapped learner be assessed to determine such things as the presence of prerequisite skills and knowledges, the current level of performance, the rate of learning and the most efficient learning style. Also specific areas of instructional difficulty must be assessed. Some of the burden of this individual assessment may be effectively handled by use of a microcomputer.

2) Prescription--Once assessment information is available it can be combined with the goals for the student to develop an individual educational plan. This IEP is mandated by federal legislation and must, among other things, include specification of behavioral or performance objectives for each handicapped student. There are already instances where microcomputers are being used in this area.

3) Formative Evaluation--There must be continued monitoring of student performance so that the educational prescription and/or the implementation process can be modified if necessary. The ability of a microcomputer to store, recall, and analyze data makes it a valuable asset in this process.

4) Resource Allocation--More and more the special education teacher is assuming the role of an orchestrator of instruction. As an orchestrator one primary function is the matching of resources with learner needs. A microcomputer can offer very valuable assistance in this area.

5) Reporting--Of all the tasks that must be undertaken by the special education teacher this is one which is frequently done the most reluctantly. The power of microcomputers to assist in collection, organization and analysis of data promises to make it a most valuable instructional management tool in this area.

INSTRUCTIONAL DELIVERY

While research has not consistently demonstrated the superiority of computer assisted instruction over regular instruction it has shown that the amount of time required for the instruction generally is reduced. This alone is sufficient to cause many special educators to consider the potential of microcomputers. There are, however, additional inducements.

1) Individualization--This is the hallmark of special education; the matching of instruction to the needs of individual handicapped learners. This is one area where the potential of microcomputers is extensive. This includes such things as: a) Self-pacing. The handicapped learner can be freed of temporal constraints posed by group instruction and permitted considerable individual control of the rate of instruction. b) Level and Sequence of Instruction. These can be tailored to student needs. By the use of flexible branching the handicapped learner can be automatically moved from one level of instruction to another as necessary. c) Reinforcement. Handicapped learners can also be provided with the type and degree of reinforcement desired by the teacher. d) Multisensory Instruction. The use of multisensory instruction is very beneficial for many handicapped learners and microcomputers have the capacity of providing such instruction through the use of sound, graphics, color, or peripheral devices such as speech synthesizers, light pens, tactile boards, etc. can be interfaced with microcomputers.

2) Motivation--A student who is motivated learns more and learns quicker than one of equal ability who is not. Not all students are equally

motivated by the same things so considerable time and energy is spent trying to identify instructional environments, methods, and materials which will have high motivational valence. There is sufficient reporting of having to "Pry the kids off the microcomputer." and having them "beat the teacher to school to use the microcomputer" to attest to the intrinsic motivation of the microcomputer for many. Novelty undoubtedly plays a role in this but so do other factors; factors recognized as important in educating the handicapped such as personalization and the ability to get, hold, and direct student attention.

3) Interaction--Possibly the most valuable aspect of instruction using microcomputers is that they permit interactive learning. Microcomputers offer a whole new avenue for making learning a participatory process for the handicapped. Through use of this medium the student can be provided immediate feedback regarding his/her response or performance. This can take the form of reiteration of information or of additional instruction. The microcomputer can prevent the handicapped student from having to wait minutes, hours, or even days to obtain feedback about performance.

It is thus quite possible for microcomputers to be a boon by providing highly motivating, interactive, individualized instruction and by facilitating the management of that instruction. However, perhaps it is best that special educators look closely before they leap for microcomputers can also be a bane.

MICROCOMPUTERS AS A BANE

Few things are entirely positive in their effects. The law of action-reaction found in the physical sciences seems to have a counterpart in the social sciences. The possible positive aspects of the use of microcomputers in special education are based upon certain assumptions. Assumptions which, if not met, can result in problems.

The possibility exists that microcomputers will become a bane to special education if their use is not appropriate for the handicapped learners. One of the factors which may cause this to occur is the software available. Without appropriate software a vast amount of the potential of a microcomputer for instruction and instructional management is lost. Those familiar with the current status of microcomputer educational software are aware of the myriad of problems which exist. The unwary can easily fall victim to purchasing software which is educationally incompatible; that is software which does not meet curriculum, learner or teacher needs.

Not only is a great deal of software educationally incompatible with special education but much of it has instructional design problems. Literature surrounding evaluation of educational software abounds with "horror stories" about instructional designs which not only hinder learning but can actually do harm. Consider, for instance, the effect on a handicapped student of using a program which responds to an incorrect response with a low toned "raspberry" and a flashing "Wrong answer you dummy!" Improper analysis of the content, inaccurate content, poor sequencing, improper reinforcement, confusing or absent instructions, no feedback, etc. are also frequently encountered.

A surprising amount of educational materials on the market today also have problems with technical adequacy. Programs are found that do not run as they should or that "crash" with unexpected types of input and, with handicapped learners, the one thing that is expected is the unexpected. Also, software which offers the teacher no flexibility or control can prove to be of very limited use in special education applications. Failure to use flexible branching, use of confusing or cluttered displays of text or graphics, etc. are all technical problems which can greatly affect the utility of microcomputer software in special education.

It is not, however only the

quality of the software which can make microcomputers a bane. The way in which they are used will play as big if not bigger role. It may be for some educators that microcomputers will be a passing fad. Some special educators will be lured into purchase of microcomputers by the "whistles and bells"--the marketing polish which demonstrates the capabilities of the hardware. They may be lured only to find that software must either be purchased or developed to make use of the special features of the hardware. And due to lack of time and money for this development, microcomputers may go unused.

Another potential problem is that microcomputers will be misused. Attempting to use this technology to solve all of the problems of instruction, for instance, will create more problems. Any method, material, or technology which is used exclusively in the process of instruction of the handicapped will inevitably be a bane. On the other hand, some special educators may see the use of microcomputers from too narrow a perspective. This can result in use of only a very limited amount of their potential. For example, a teacher who only uses a microcomputer as an "electronic page turner" or as a "drill master" may not be using it in a manner which justifies its expense.

There are many possible abuses of microcomputers in special education settings and such abuses could cause microcomputers to become a bane in special education, just as may the presence of inappropriate and ineffective educational software.

Will microcomputers be a boon to special education or a bane? The answer to this question lies with hardware manufacturers, software producers, teacher preparation institutions, school administrators, and special education teachers. With proper programming and use they will be a great boon. Without these they will be a definite bane.

27J

"Transparent Sorts Better Serve Users"

Presenter: William R. Muldrow
Department of Education
Management Information Services

Introduction

The problem of report sequence is one of the most commonly occurring problems for educational administrators and other data users. The major objective of an application to this problem is to provide reports for the user in the order and sequence desired by the user. Normally, chronological ordering and sequencing of data are dependent upon computer assigned values or utility software assigned values. Often, these chronologically assigned values do not adequately satisfy the user's needs.

The problem is complicated by the need to honor varying requests, and to satisfy many local and external governmental requirements, coupled with the many public interest inquiries.

In its most general form, the sort or sequencing problem encompasses the following requirements:

1. Arranging, organizing, and developing portions of files to accommodate data which will be transparent to the user in subsequent reports.
2. Developing a record processing interface program to arrange, reorganize, and reformat data to the user's specifications.
3. Incorporate a utility sort in the job stream that can be user controlled.
4. Whenever possible, satisfy the preference of the user as to output report sequence requirements.

Solution Approaches

At the onset, it should be understood that each report requiring the special treatment of user specified sort sequence that does not adhere to chronological value assignments of characters should be developed from a sequential file copy of the original file. However, this does not necessarily require a permanent hard copy file. A temporary file can be created for the job run only. The temporary file copy method is the most preferable, but may not be an available routine for some small machine users, except where a programmer's file creation and scratch routine is developed for utilization of an off-line device. The file copy, whether temporary or of a permanent nature, will allow the application's programmer to fill the transparent storage area with new data to the user's specifications relative to any given report. This process may be accomplished as shown by Transparency 1. As can be seen from Transparency 1, the application's programmer must modify the data to the user's specifications and store such data in the blank field reserved for this operation. An example of a typical application of modified data is as follows:

> The user wishes to report "St. Mary's School" as "Saint Mary's School" and visible to him as "St. Mary School." The application's programmer must test the original data area for "St. Mary's School", and for each find, he will replace the data (once moved in the working computer memory) with the user's specified sort version of the data; that is, he will change "St. Mary's" to "Saint Mary's" and place the latter spelling of the data in a file copy as shown by the generalized demonstration entitled Transparency 1.

The process is repeated until as many desired modifications can be made and stored, however, limited to the provided storage space in the reserved area (transparent area).

Once the data processing is completed and the copy file is stored, the application's programmer should terminate that step for output use in a subsequent utility sort step. Once this procedure is established, it becomes obvious that utility sort incorporated in the job stream is the most efficient software organization because it allows the application's programmer to arrange the file in the desired sequence before any reporting is done, that is, in reference to the job run.

As a final step in the transparent sort development the programmer should define the temporary file developed in the prior step as the input file for the report generator. By doing this, the programmer has the option of using any language to develop the desired report, since the record output will be an I/O routine only. This does not mean that record processing can't take place in this step, it simply means that programmable sorts will not be allowed to accomplish the final step of the transparent sorted output.

Model Development and Usage

An applicable model was developed by the author of this paper to satisfy some administrative data needs subsequent to the mandated development of the FIRN (Florida Information Resource Network) plan by the Florida legislature during the 1981 regular session.

Department of Education's administrators, charged with and delegated the responsibility for implementation of the FIRN plan concept, needed a quick, automated approach for report analysis and information references of data collection by the Department. Out of this need grew several problems which could not be solved adequately and quickly with conventional programming and systems techniques. These problems are outlines as follows:

1. Each of the data collection forms utilized by the Department had to be analyzed.
2. Data types for each of the forms had to be defined and characterized.
3. Data types and relevant characteristics had to be recorded and reported in a timely manner for ongoing planning information.
4. Recorded data types had to be researchable and retrievable in a quick and preferable format.
5. Report sequence was in a nonstandard form.
6. Form analysis, data file development, and specified report production capability had to be completed in approximately ninety days.

Model Development Activities

1. The report format, sequencing, and method of production was determined.
2. Data entry and file development procedures were developed.
3. File editing techniques were developed.
4. Report generator software was developed.
5. Report formating was completed.
6. Sample files were developed from actual records.
7. Sample reports were run from sample files.
8. Reports were generated from the entire file.
9. Final development of application and system software was done.
10. Successful report runs were made.
11. Development of software documentation and run specification was achieved.

Model Design

A general understanding of the problem and a model solution can be graphically shown by Transparency 2. The first step of the design is to develop a data entry process and procedure. If the data entry process is offline to the main processing system where final processing of the product will be done, it may be desirable to create a compatible input medium for the main system for that device. The model described in this paper, as shown by Transparency 2, used an off-line system for data entry processing. However, some institutions will only have card decks available as a compatible medium to other systems. In that case, the system developer will replace the data entry step shown in Transparency 2 with a card entry or card image data source.

The second major step of the model is a copy and catalog procedure to establish a residence cataloged, public (to the main processing system) data set. However, this may be any medium common to the main processing system. The creation of this set should be to establish a permanent data set anticipating no changes to that set during the particular data use duration. The general structure of the set should be a design with the same DCB parameters as the input set from data entry.

The steps that follow the creation of the master file processes the sort fields in each record to obtain field data formats to the user's specifications. Transparency 3 shows a sample report from the unmodified file. Notice that Transparency 3 shows such statements as "INSTRUCTIONAL STAFF SERVED" preceding the field statement "Instructional Staff Area/State Inservice Workshop." The user analyzed and defined these statements to the author of this model and paper as follows:

> "Instructional Staff Served" as a complete statement with a higher sequential value than "Instructional Staff, . . .", that is, since the information following the comma was intended only as more descriptive for the "Instructional Staff" look-up.

In other words, to the user the physical view of the data should look as follows:

> Instructional staff served
> Instructional staff

The user only wished to enter such statements as

"Instructional Staff, Area/State Inservice Workshops" to fully describe the entry preceding the comma.

In the sample problem cited above where "Instructional Staff Served" shows a blank following "staff" and the blank being of lower value than a comma which follows the other example. The sort is produced in reverse order of the user's preference.

Other user preferences and specifications were also subjects of the design criteria for the model. Some user concerns were as follows:

1. Sentences or statements ending in periods with other detailed information following.
2. Excessive blank spaces in data string.

Each data format problem was solved in a record processing step, a utility sort was run against the processed temporary file and a new temporary sorted file was created for reporting.

Notice Transparency 3. There are two data output statements in a sorted order, "Time Required, DMTS . . ." and "Time Required Estimate, DMTS . . ." In an ordinary sort sequence, the first of the two statements would appear last because, in the first a comma follows "Time Required" and in the latter a blank follows "Time Required." The blank being of a lessor value than a nonblank character would cause the latter to appear first in any ascending sort sequence. However, with the modified data transparent sort the reverse order output was the result specified by the user. The same modified sort trend can be seen in Transparencies 4, 5, and 6.

Transparencies 7 and 8 are outputs or "snapshots" of data during several stages of processing. Notice that each statement appears four (4) times. The first occurrence of each statement is unmodified, and the latter is a completion of the modification. It also can be observed that the comma, period, and excessive blanks are removed during processing. The original file structure of this data was an organization in three (3) fields as shown in Transparencies 4, 5, and 6. Therefore, the process can be summarized to this point as:

1. The data from the related fields were placed in a single string and field.
2. Left and right parenthesis were removed.
3. Excessive spaces were removed.
4. Finally, the commas and periods were removed.

The ordering and sequencing of the above steps are important to accomplish exactly what was intended at the onset of the development of this model.

The final major processing step before executing the utility sort routine was to store the complete modified data string in an unutilized file storage area, that is, a portion of the file described as blank spaces or unutilized. This process is graphically demonstrated by Transparency 1.

The utility sort step was and should be kept very simple. In the model, the entire string was defined as the only sort control for an ascending fixed output format. The sort simply arranged the records in ascending value order, and an output file was specified. Again, the output file was defined as a job run, temporary or scratched file since the file was needed only for a particular job run. This caused the current master to remain in the systems record format.

As can be observed, from the procedures outlined in this paper, utilization of the model is application oriented. Therefore, different procedures must be developed to satisfy user specificaitons. This does not mean that the procedure can't become fully operational, but it does imply that the modification step must be unique to each job description placed in operation for frequent production runs.

Implications

Several implications appear to be apparent in the transparent sort sequence cited by this author. However, the most significant implication is that it is an approach to providing a product that serves the user's interest best. Often, users specifications as to the exact order and sequence of report data are not parallel to computer system specifications as to the sequencing or ordering of the same data outputs. Also, as can be implied from this paper, systems or utility sorts are usually more efficient than applications programmed sorts.

The user needs and specifications can be better satisfied through the use of transparent sorts. By this, it is intended that the application's programmer must make an assessment of the users needs and specifications as to automated reporting, that is, in terms of order and sequence. Secondly, to satisfy those needs when it is determined that a standard utility sort will not provide the order and sequence desired by the user, a record processing step must be developed to provide a sort control file area transparent to the user. Literally, this does not mean that the area is provided at the time, but it is filled in with the modified data at that time. The transparent sort concept must find its first, prerequisite design with the foresight of the programming and systems group who must first develop the original file structure, machine records format, and data file. Inherent in this concept and procedure, for most efficient use, is a provision for enough blank space in the original file to provide space for anticipated, modified data as to a user's specifications. However, this provision, coupled with available file space, may cause some limitations as to the number of second, third, fourth, and other level sorts when desired by the user. Available file space may be the most significant limitation as to implementaion of the transparent sort concept.

One major plus for the transparent sort concept is that the number of reports desired by the user with conventional sorts may be reduced by making modifications to simplify the look-up for any given report. Often, multiple reports consist of the same data, but sorted in different ways to allow easy reference to material.

One of the most limiting factors of the procedure described in this paper is that an application's programmer is a necessity for effective implementation. This is not a procedure that contract application-development should be considered as a feasible solution to the problem, since constant user input is a requirement for producing the desireable transparent sort. Further, the user should have the option to make cosmetic changes to improve the desired sort. This is done only with an application program step of the job or procedure. However, this does not eliminate entirely the contract option, but if such an option is selected, the user will have to decide on each output option in advance of the issuance of contracts. Again, contract programming may not be a very economically feasible solution since the procedure usually requires constant interaction between the program developer and the user to assure the desired results. When the output report results are to be utilized for publications such as directories and other reference material, the service contract method of development may be cost feasible for the user. Also in such cases, the time spent with the application's programmer to develop the application may also be a feasible investment.

Integrating Microcomputers into the Mathematics Curriculum--Using the Computer as an Adjunct Instructor.

The Iowa Microcomputer Curriculum Project was begun with the goal of making the microcomputer accessible as an adjunct to the contemporary classroom. The project is just completing its first year of operation, Phase I which focuses on algebra I, geometry, and algebra III.

The project was funded for 1981-82 with Title IV-C funds administrated by the Iowa Department of Public Instruction. Project activities include six distinct but interrelated areas of activity that provide user friendly, quality programs at low cost to educators.

The first task to be completed was the development of a content module - skills development matrix. The content module dimension of the matrix identifies content modules of short duration that are common to the majority of algebra I, geometry, and algebra II curriculums. The skills development dimension of the matrix identifies three stages of concept development that are believed to be compatible with the use of microcomputers. The initial matrices were developed in-house and then submitted to a panel of external curriculum specialists for critical review. The matrices and the critiques were then reviewed and the matrices were revised by a local committee. All of the suggestions that had been provided by the external specialists were considered and most were incorporated into the revised matrices. A portion of the algebra matrix is shown below.

ALGEBRA I	Concept Development Stages		
Concept Module	Introduction (I)	Skill Building (S)	Application (A)
7 Graphic solution of single variable inequalities	Disk AI-1 Graphing One-variable Inequalities (Introduction) C. Jones		
8 Absolute value		X	
9 Graphing in the rectangular coordinate system: a) given rules of the form ax + by = c make a table and graph	X	Disk AI-2 Rule to Tables to Graph (Skill Building) R. Eggert	
b) transform to y = mx + b form then graph	X	Disk AI-2 Transform to y=mx + b and Graph (Skill Building) J. Gallagher	
c) given the slope and y-intercept write the equation and graph	Disk AI-2 Slope/Intercept to Rule to Graph (Introduction) G. Rasmussen	Disk AI-2 Slope/Intercept to Rule to Graph (Skill Building) J. Cowley	

The second task included the design and production of two one-week workshops for author-programs. Forty experienced teachers of algebra I, geometry, and algebra II with proven programming skills were recruited to participate in an intense programming experience. Participants were provided a one hundred dollar stipend for living expenses and tuition for the course. All participants were housed in a university dormatory within walking distance of the computer laboratory established for the workshop. Most participants were able to borrow a microcomputer from their school districts.

Consultants from the Minnesota Educational Computer Consortium provided a one day intensive workshop on design and development of software. Other consultants spoke on the topics of structured programming, error trapping, restricted input and screen organization. Each participate selected two matrix cells as the focus of development. The participants efforts were applied to the program for one of the cells during the workshop. The second program was to be completed after returning home from the workshop.

The workshops were successful in that eighty programs were produced. The programs submitted required additional further processing before being distributed to schools.

Program processing and refinement dominates the third major task. To date, all processing has been done by project staff. Programs are received on 5½" diskettes in various stages of acceptability. Each program is reviewed by the Project Manager who then directs the staff to make minor revisions before the program is rereviewed by members of the Editorial Panel. The comments provided by the Editoral Panel are then incorporated into the program and it is sent to professional referees for closer scrutiny. The referees comments help determine additional program modifications. An additional review by the Project Director and Project Manager are completed before a program is finally released.

The fourth task involves the development of a delivery system. Finished programs are placed on a distribution disk that is user friendly. The disk requires no computer expertise. The user inserts disk into the drive and turns on the computer. A disk title page appears and is followed by a descriptive menu that is single key actuated. The user presses the letter corresponding to the desired program and the menu does the rest. A program title page is presented that provides the program title, authors, name and authors school. The program is then run under user control. When the user is finished with the program, the disk menu is presented again.

A fifth task addressed by the project has been the development of a user workshop. The user workshop assumes no prior computer experience and focuses on various ways in which programs from each concept development stage may be used by the classroom teacher. The very personal matter of classroom management and other appropriate details are discussed. A card has been prepared that provides detailed instructions for starting the system and turning off the system.

The sixth task of the project has been that of disseminating information about the project and its products and distributing the software it produces. The project publishes micros in Mathematics Quarterly (MMQ) which contains current listings of available programs. MMQ also lists matrix cells for which programs have been received and are in the process of evaluation and/or revision. One of the other features of MMQ is a short profile of the authors of programs which have recently been added to the distribution matrix.

Project staff members attend state, regional and national conferences and make presentations about the project. The project diskettes are available at low cost to school buildings through a purchaser agreement included in each MMQ.

Each diskette is internally coded with a serial number and the purchasing school name. Each purchasing school has the right to make additional copies for use within the building for which the original software was purchased. Other copies are prohibited. The software is offered to individual buildings on a per disk basis or through intermediate school agencies on a contract basis.

A user feedback loop has been designed that will allow further program improvement based on user suggestions.

Plans for the second year of the project include continued development of programs for the current matrices however, the project will be expanded to include middle school matheamtics. The Phase I model described above will be modified to fit the special needs of the middle school mathematics curriculum. A matrix will be developed with a grade by grade organization but with a topical strand flavor. Software primarily designed for use in grade six may serve very well in grade five for one purpose and also in grade seven for a different purpose.

The authors workshop will be replaced with a designers workshop. The participants will study design problems peculiar to microcomputer programs.

They will be chosen from middle school curriculum specialists. No programming expertise will be required. Each designer will select at least two cells from one of the middle school matrices. They will then specify the desired objective and the prerequisite skills required to reach the objective. A strategy will be developed to enable the students to obtain a satisfactory mastery of each skill level.

An explicit "story line" complete with individual screen layouts will be developed for each program. This detailed description of the activity will be converted to a computer program by a team of programmers. The completed program will then go through the same evaluation process developed in Phase I.

The designer workshop model has important implications for the planned expansion of the project to other curricular areas. It must provide a means by which curriculum specialists who are nonprogrammers can participate in providing meaningful computer activities within the framework of the project.

The development of an adequate design model will allow the project to be expanded to include most curricular areas at all grade levels K-12.

- Lynn C. Schwandt

Managing a Major Resource - Management and Instructional Information

Brenda Somers

Divisional Assistant, St. Louis Public Schools

INTRODUCTION

From one point of view, according to Morton Meltzer (1981), every person within the spectrum of management is a manager of information. From the other side, sich great demands are made on every administrator to manage information that there is a requirement within the heirarch of the organization for some one person to be responsible for the coordination of information resources and information requirements.

One comment made by Meltzer should be kept in focus as one assesses this presentation:

> "Failing to manage information or mismanaging it can cut into an organization's profits by increasing operating expenses. Some of the direct results of mismanaged information are duplication of effort, decisions based on outdated or erroneous information and decreased productivity." (Meltzer, 1981, p.6).

The Division of Information Support Services, St. Louis Public Schools

The Division of Information Support Services was established in July, 1979, as part of the St. Louis Public Schools' Direct Delivery System and the concomitant reorganization. It is one of three parallel divisions which makes up Data Management Services. The other two are the Division of Data Processing and Evaluation and Research. The Division of Information Support Services (I.S.S.) is charged with the responsibility of providing and monitoring information for St. Louis Public Schools personnel involved in educational decision-making. Specifically, the personnel in the Division of I.S.S. act as a clearinghouse for the collection compiling, storing, retrieving, and delivering information to management and instructional personnel. The precipitating factor were the needs to reduce data redundancy, to reduce duplicate data collection tasks required by principals, and to design computerized data systems for the collection, storage and access of data.

With efforts to serve the ultimate goal of improving the quality of education for our students, I.S.S. provides services which requesting clients may obtain, such as the preparation of specifications for the design or modification of major data systems; consultation on the development of data collection activities, such as surveys and needs assessments; and production of ad hoc computer reports on available computerized data such as student records, achievement scores and personnel data.

The Division of I.S.S. also acts as the archives for selected data processing reports, Board minutes and reports, evaluation reports, and state and national statistics.

Viewing information as a corporate asset and a tool for decision-making, I.S.S. provides a variety of services which are described in the next section.

FUNCTIONS

In order to assign tasks to the seven professional and seven clerical staff members, as well as to communicate functions to field users and central office users, various functions have specific names as if they were units of the divisions, although in some cases functions are shared among the members of the division. For instance, each divisional assistant, the professional staff members, complete system analysis in their particular assigned areas: student accounting, personnel, financial, and facilities accounting.

1. Data Collection Clearinghouse: the goal of this function is the improvement and/or elimination of data collection instruments and surveys. I.S.S. provides the review, authorization and monitoring of data collections within the system as well as those requested by external researchers. Control of redundancy is a major aim. A Master Calendar for system-wide activities and a monthly calendar of events and reports due are included in this function.

2. The Information Center stores, abstracts and retrieves corporate documents and provides searches for relevant research through ERIC and Educational Research Service (ERS).

3. Ad Hoc requests for information are sent to I.S.S. when a need for a one-time non-standard report of data already collected is needed. Using ASI-ST report-writer software, labels, lists, and aggregated counts are provided to clients. Statistical studies are also provided using SPSS or SAS.

4. The Division of I.S.S. is involved in Data Base Management in conjunction with Data Processing Data Base Analyst. The DBM coordinates the specifications for the logical design of the data base and corporate needs for the inclusion or exclusion of data elements.

5. System Analyses are provided by various members of the professional staff so that each major system has been documented and recommendations for changes can be made. Student accounting (which includes achievement), Personnel, Facilities, and Financial systems are the major focus. Flowcharts, data directories, and data dictionaries are included.

6. Documentation of both the input and output of the major systems is also provided. User's manuals and explanations of CRT displays are also provided to allow the users to become more independent.

7. Staff Development of users is a major function, again encouraging user independence. Courses are provided three times a year on the available software (such as ASI-ST, VOLLIE, SPSS, SAS, MANTIS, OETC) and pertinent concepts (Data Base, Structured Analysis, Survey Research).

8. Liaison to the field is focused on the area or sub-district offices. In this way, each of the 40 principals and staff personnel of each area have a contact person and I.S.S. can be well-informed of the field's need for information.

9. Title I Data Management is charged with providing the data required by the Missouri Department of Elementary and Secondary Education. This includes data to support which students are eligible, who is being served and how long, comparability data, and planning and state application information.

SUMMARY OF ACTIVITY

The functions of the division described should be placed in perspective to the size of the organization.

The enrollment of the St. Louis Public Schools in the three years that the Division of Information Support Services has been in existence has been approximately 60,000. Although some schools have been closed, others have been opened as a result of movement of geographical lines for base schools and the change from a K-8, 9-12 configuration to include middle schools (K-5, 6-8, 9-12) in 1980 for desegregation purposes. There generally have been 100 buildings which house students Grade 8 or younger and approximately 19 secondary and 6 special education schools for a total of 126 schools.

Four area (or subdistrict) offices have been in existence since 1979 (prior to that there were five). Three of the subdistricts are geographical in nature, dividing the district's 61 square miles into three north-south sectors. The fourth subdistrict is not geographical, but is tied together by programs: all magnet schools and alternative schools.

We have 4,496 teachers and administrators (certificated) and 2,735 classified or support personnel for a total of 7,231 staff.

With these statistics as a basis for the clientele for the Division of Information Support Services, the capabilities of a small division in one school year can be put into some kind of perspective. One of the difficulties in ascertaining need and value of services is the calculation of time. For instance, for several weeks, all seven professional staff members were working on one major project, a report to the state department, on request by another division. This kind of effort may not be apparent in output statistics presented in the following paragraphs.

The first set of figures shows the requests by office for 1980-81 and the time in hours as recorded.

OFFICE	N	TIME (HRS)
Central	301	1182.00
Area I	96	129.50
Area II	77	58.50
Area III	100	122.16
Area IV	69	94.33
Vo-Tech	2	1.00
External	94	226.33
Totals	739*	1813.82

*This total only represents requests that were recorded. See ad hoc requests.

The next table shows the number of requests by month from June, 1980 to May, 1981

MONTH	NO. OF REQUESTS
June, 1980	62
July	37
August	31
September	51
October	89
November	58
December	56
January	98
February	57
March	91
April	64
May	45

In addition to information requests I.S.S. reviewed 200 data collection instruments during the 1980-81 school year. More than 140 were redesigned before being approved. Redesign included revision of instructions, rewriting items and deleting redundant or unnecessary items. About 45 instruments were either rejected or merged with existing instruments.

Some ad hoc requests for information were handled as 5 minute telephone calls or walk-in requests, and were not recorded. In one two-week span ad hoc requests were recorded and I.S.S. handled 78 such requests. Extrapolated to a full year, this would total over 2,000 ad hoc requests.

For the administration of the Title I roster process, each month Title I teachers and principals make hundreds of calls to I.S.S. to request information on specific problems or on specific students. The time to take care of one request could range from 1 hour to 1 minute. The teachers and principals enjoy the access they have to someone(s) who can give them specific instructions on how to properly provide to us the necessary information about the Title I services in their schools.

SUMMARY AND CONCLUSION

Although this article is but a brief outline of the services that one organization provides its members, the need for control and easy access to corporate information in a school system must be stressed. In times of critical economic strains, more than ever management decisions must be based on relevant, timely, and accurate information. As such information is a resource not unlike personnel and facilities, all of which have a cost to obtain, maintain, improve and utilize to their potentials. The methods of accomplishing these objectives are a function of the organization and its primary goals.

References:
Meltzer, Morton F. Information: the ultimate management resource
New York: Amacom, 1981

Computer Literacy in the Elementary School: An Argument for Change from Within

Alice-Ann Winner

Although computers are used in endless variety in most aspects of modern society, the history of the computer spans a mere four decades (Evans, 1979). Educational computing has had an even shorter lifespan, only twenty-five years (Moursund, 1980a). The major focus of this first quarter century of educational computing has been at the secondary and/ or post-secondary level; elementary education, for the most part, has not felt the impact of the computer until recently.

This exclusion was due in part to the complexity and expense of the early models. They were kept in separate areas, under the care of specialists, and did not become part of classroom regularities. In addition, the first descriptions of computer use spoke of possible teacher replacement causing negative reactions to the concept of computers in education.

Microcomputers, however, have changed much of this early negation. The introduction of the microcomputer into education, made possible by silicon-chip technology, has implications of revolutionary proportions for elementary education. The computer has moved from the world of the experts to the world of the elementary school. Microcomputers are less costly, more manageable and sturdier machines which easily fit into the classroom. Both teachers and children can learn to load them with minimal instruction. The interactive mode is well-suited to the needs of elementary children, providing a high degree of motivation (Winner & McClung, 1981; Winner, 1982). Microcomputers are fast becoming THE machine to purchase.

Instructional technologists warn, however, that rapid purchase of equipment without complimentary curriculum development can lead to misuse and swift obsolesence (Winner & Holloway, 1982). Much thought needs to be given to both the question of how microcomputers are to be used in the classroom and to the task of developing computer literacy and compentency in teachers. What exactly do elementary teachers need and want to know about computers in order to begin using them in their classroom instruction?

The need for training in intelligent computer use is apparent, as it is the classroom teacher who will have the final say on the use of computers in the elementary school. It would be a misuse merely to add the microcomputer to the current curriculum. Add-on curriculum innovations have been attempted before with poor results (Winner & Holloway, 1982). As an add-on to the existing curriculum there is a strong possiblity that the computer will not realize its full potential. Instead of becoming an instrument of profound educational change, it may instead become only another short-lived technological innovation.

The recent literature supports the view that a restructuring of the curriculum will be necessary to exploit the potential of the microcomputer (Moursund, 1980b; Molnar 1976, 1980a, 1980b). Curriculum revision, however, is difficult to accomplish. It requires both a knowledge of the factors involved and a commitment to attempt the necessary changes.

Change in Education

Significant and lasting change is a rare phenomenon in education, especially at the elementary level (Bredo & Bredo, 1975). In the twenty-five years since Sputnik, many innovations, such as the "new math", programmed instruction, educational television, open classrooms and language labs, have not been able to sustain their initial promise.

the use of computers must keep in mind the perspective from which educators come" (Rawitsch, 1982, p. 31). This is especially significant in the training of elementary teachers as they tend to be less confident in the fields of mathematics and science from which much of the computer language and logic are derived. Linkage of computer technicalities to concepts familiar to teachers, omission of extraneous information,and the use of organizing models to give teachers a grasp of the total computer picture are thought to be positive methods of attacking the inservice training problem (Rawitsch, 1982).

Towards in Inservice Model

At the United Nations International School in New York City, we are engaged in developing an inservice training model which could serve the needs of elementary teachers in their quest for computer literacy and compentency skills. Using the change literature on innovation diffusion as our guide, we have implemented a series of workshops for our teachers with the ultimate aim of curriculum revision through microcomputer technology.

UNIS possesses some of the essential attributes described in the change literature. Its diversity of culture and educational philosophies creates a flexible atmosphere sympathetic to change.The faculty has a strong voice in curriculum development. The principal is supportive and politically astute.The staff's attitude toward themselves and the children they teach is positive, and mirrors a sense of commitment to continuous development. This atmosphere of flexiblity and openness produces a climate in which innovation adoption and risk is possible.

Our aims for this inservice program were both modest and ambitious. Our short-term goals were to increase faculty awareness of computer use in the classroom, extending this awareness to exploration and experimentation of computers both individually and with the children. Our long-term objective is to augment curriculum development and spark a renaissance in teaching and learning with the computer as an instructional aide to this process.

The three aspects of the diffusion literature thought to be advantagous for implementing lasting change were used in the program design, implementation, and evaluation. Computer use and attendance at the workshop sessions was completely voluntary. The change agent was a staff member, a fellow teacher, who designed and implemented the training program, in addition to providing supportive assistance in the intervals between sessions.The workshop series consisted of thirteen training sessions held twice monthly throughout the current academic year, thus allowing ample time for gaining confidence with the complexities imposed by the computer.

The inservice program series used a formative evaluatory process to secure continuous feedback.The course content was adjusted to reflect the on-going requests, needs and desires of the participants. Approximately one third of the teaching staff of our elementary school participated on a regular basis. The feedback has been enthusiastic and spirited, and our six microcomputers are now receiving wide experimental use.

The diffusion literature supports our view that change from within has more promise of fruition (Paul, 1977). Although it is still too early to access our long-term objectives, our short-term goals have been realized. The level of discussion in our workshop series has been stimulating and productive. Our teachers shared their beginning experiences, both positive and negative, their hopes and anxieties, and their general insecurities in approaching change in their classroom regularities. The curriculum analysis emerging from our regular interaction has established an awareness of the potential need for change and has strenghtened both our understanding

These innovations became little more than educational fads-enthusiastically greeted at the outset, accomplishing little substantial instructional advantage during their brief adoption period, and ultimately receiving only marginal use or discontinuence (Rogers & Shoemaker, 1971). The typical classroom returned to its traditional norms.

This dearth of educational advantage in the change history of the post-Sputnik era has several causes. Often innovations were adminstrative decisions imposed upon teachers with insufficient discussion and inservice training. Or they were implemented without complementary field-testing procedures, and proved to be inadequate to the task. Or the innovations were developed by outside experts who, though knowledgeable in their particular disciplines were ignorant of classroom priorities, and developed materials which conflicted with exisiting contingencies.

Change is a process. It is not merely the implementation of a product or the addition of a concept to the curriculum, and it needs to be understood in those terms if change is to be lasting. Research has illuminated several important generalizations requiring careful study by those who wish to secure innovation adoption and implementation. Briefly stated, change is more permanent if the following situations exist:

1) the innovation decision is participatory rather than authoritarian - the teachers themselves decide that the change is advantageous and necessary;

2) the change agent is familar with the procedures, social climate, and contingencies of the organization where the innovation will occur, and that he/she is similar in characteristics and acceptable to the participating teachers; and

3) ample time is given to the innovation process, and continuing support and reinforcement mechanisms are provided.

Teacher Training and the Change Process

Given the importance of computers to the future of education, it is essential that elementary teachers receive adequate training in all aspects of microcomputer use in the classroom. Appropriate inservice training is a necessary prerequiste and this training should be sensitive to our understanding of change as process for optimal computer implementation.

Most of the present computer workshops for educators do not meet the needs of elementary teachers. They are conducted by specialists with only marginal knowledge of the complexities of elementary education, and there is little connection between the course content and the real world of the classroom.A rather heavy dose of computer jargon and technical complexities are provided. This tends to raise the level of difficulty to significant proportions. The perceived difficulty of an innovation is adversely related to the degree of its adoption (Rogers & Shoemaker, 1971). Too much technical expertise might prove to be counterproductive.

Elementary teachers' initial literacy and compentency experiences will color their view of computer usefulness in education.If the training is irrelevant, too complex or technical, educators will shy away from mastery. Computers will be utilized in the classroom in their most inflexible but most easily implemented mode--CAI drill and practice programs, and become little more than electronic workbooks. If we are to see the flowering of education through creative use of microcomputers, literacy and compentency training must be tailored to the specific needs of the users.

The training of teachers in computer use has received scant attention in the literature (Rawitsch, 1982). It is different from the training that children would receive or that of computer professionals'. "Instructors involved in training educators in of the involved complexities and our commitment to move in that direction. We have begun to openly discuss some facets of curriculum revision and a desire to develop software for our own specific needs.

While we are aware that there exists a large chasm between talking about change and its actual implementation, permanent change will not occur without this preliminary discussion and commitment. We are expending our energies in this direction in the hope of providing a strong foundation for the curriculum innovations we believe to be a fundamental aspect of wise computer use in the elementary school.

References:

Bredo, A.E. & Bredo, E.R. Effects of environment & structure on the process of innovation. In J.V.Baldridge & T.E.Deal (Eds.) *Managing change in educational organizations*. Berkeley: McCutchan, 1975.

Moursund, D. Preface. In R.P.Taylor Ed.) *The computer in the school: Tutor, tool, tutee.* New York: TC Press. 1980.

-----------. *Teachers guide to computers in the elementary school*. La Grande, OR.:ICCE, 1980

Molnar, A.R. The use of computers in education. *THE Journal*, 1976, *3*(1), 18-23.

------------. Understanding how to use machines to work smarter in an information society. *THE Journal*, 1980, *7*,(1), 42-47.

------------. Microcomputers & videaodisc: Innovations of the second kind. *THE Journal* 1980, *7*,(4), 58-62.

Paul, D.A. Change process at the elementary, secondary & post-secondary levels of education. In N.Nash & J. Culbertson, *Linking processes in educational innovation*. Columbus, O.: NCEA, 1977.

Rawitsch, D.G. Teaching educators about computing: A different ballgame. *The Computing Teacher*, 1981, *9*(4), 27-32.

Rogers, E.M. & Shoemaker, F.I. *Communication of Innovations*. New York, The Free Press, 1971.

Winner, A.A. Elementary problem-solving with the microcomputer, *The Computing Teacher*, 1982, *9*, (6), 11-14.

Winner, A.A., & Holloway, R.E. *Technology integration for a new curriculum*. Paper presented at the meeting of Mid-South AEDS, New Orleans, February, 1982.

Winner, A.A., & McClung, M.D. Computer game-playing: A 'turn-on' to math. *Arithmetic Teacher*, 1981, *29*, (2), 36-37.

WHAT SHOULD BE INCLUDED IN COMPUTER LITERACY TRAINING FOR TEACHERS?

Bernice Doerr Andrews
University of Central Florida

The pervasiveness of computers in our society is undeniable. Computer applications and advertisements beset us in every form of the media every day. In this information/computer society, where information is power and further differentiation between the rich and the poor hinges on information, there is a growing recognition of the importance of widespread public understanding of computers (Anderson and Klassen, 1981). Consequently, computer literacy has been referred to as "the next crisis in American education" (Molnar, 1978).

Various definitions of computer literacy have been proposed. Some definitions are myopic in scope for they equate computer programming with computer literacy. A more comprehensive definition is necessary. According to Anderson and Klassen (1981) computer literacy is "whatever understanding, skills and attitudes one needs to function effectively within a given social role that directly or indirectly involves computers". With this definition in mind eight major topics for computer literacy objectives were proposed:

(1) Applications: includes how computers are used in society and how they can assist people.
(2) Hardware: includes the evolution of computer hardware and basic hardware terminology.
(3) Impact: includes computer occupations and the social effects of computers.
(4) Limitations: includes what computers cannot do.
(5) Programming/algorithms: includes the ability to read, debug, modify and construct algorithms and programs.
(6) Software and data processing: includes the terminology relevant to software, information and data processing.
(7) Usage: includes how to use a microcomputer or computer terminal.
(8) Values and feelings: includes developing positive attitudes toward using computers and values their role in society.

Although the definition of major topics for inclusion in a computer literacy curriculum will vary between curricula, the specific objectives are usually subsumed under different headings.

A variety of models for computer literacy teacher training have recently been proposed. These models address the question of how much teachers should know about computers and their application to education. Unfortunately, a definitive answer does not exist. Computer technology is changing at a very rapid pace. Thus, computer literacy training is an ongoing process which will never be complete.

Since any attempt to incorporate computer literacy into the K-12 curriculum depends upon teacher training, the starting point for teacher training should be the objectives of the computer literacy curriculum. The depth of teacher training must go beyond the student curriculum in order for teachers to adequately field students' more challenging questions. Teachers should also be aware of the changes in computer technology which surround them, and continuously incorporate these changes into the curriculum.

Certain topics require additional emphasis to meet teachers' unique needs. Since some school districts have school based budgeting and others may not have a Computer Education Coordinator, discussion of hardware should include the rudiments about purchasing and using equipment. Current funding necessitates the prudent expenditure of monies. Consequently, it is imperative that teachers be able to intelligently discuss alternatives with computer salesmen. In the past, some teachers have been lured to purchase minimal computer configurations only to find them insufficient shortly thereafter. Proper initial equipment configurations can eliminate hidden labor charges at a later date. Basics on how to setup equipment and use utility programs should also be considered.

A teacher's computer literacy requirements include topics not addressed in the computer literacy K-12 curriculum. Knowledge about computer applications in education and how to obtain information concerning computers in education is imperative.

Teacher training should include instructional and administrative computing applications in education, however emphasis should be placed on instructional applications such as tutorial CAI, drill and practice, and simulations. "Hands on" experience with various instructional applications is essential. Although few teachers will have the time, opportunity or inclination to develop instructional software, considerations for developing instructional software ought to be explored. This knowledge will prove to be invaluable when evaluating software.

A plethora of educational software is under development. As with anything, the quality of software varies greatly. Thus, evaluation of software is an important topic. A number of different evaluation forms should be carefully scrutinized. What evaluation criteria were used? Why? Activities of clearinghouses for software evaluation such as MicroSIFT should also be examined.

Since educational computing is a rapidly changing field, teachers need to know what information sources are available. Books, periodicals, newsletters, professional societies, hardware vendor groups and computer based bulletin boards can provide timely information. Some states publish an educational computing newsletter. In other areas, special interest groups of computer using educators have formed separately or as a subgroup of a local computer club.

Each teacher should have exposure to working with educational software tailored to their level and subject-area specialization. Experience with a wide variety of tutorial systems, games and simulations in their specialization is indispensable. Such experience may suggest other more appropriate applications which can be developed.

The computer is a powerful tool which teachers can put to use in their daily activities. It would be unfortunate to overlook some of the time saving applications available. Word processing can be used to produce handouts, tests, and reports. Numerous programs exist which can be used to analyze test data, calculate grades and class statistics. Tracking systems for student progress can be used in a variety of situations such as the remediation of basic skills and reading.

Computer literacy takes on an additional meaning for teachers of computing (e.g. computer mathematics and business data processing). Many teachers of computing are forced to function as a general resource to the administration, faculty, and students at their school. It is not uncommon for a computing teacher to develop and administer programs for attendance, library fines and sports statistics. Consequently, their knowledge must go beyond the mere basics. Unfortunately few states certify computer science teachers. What other subject area is taught which requires no content courses?

Teachers of computing should possess the basic competencies needed by all computer professionals. The Curriculum Committee on Computer Science of ACM has recommended the following basic topics: knowledge of programming, software organization, hardware organization, data structures and file processing. Knowledge of how to program does not assure that one can teach computer science. Topics relevant to teaching computer science must also be addressed. Application of learning theories to learning about computers is critical. Examination of the scope and sequence of existing curricula for different computer science programs allows teachers to explore and evaluate various alternatives.

If the development of a computer literate society is to be achieved, an important first step is the training of computer literate teachers. The following recommendations should be strongly considered for the effective implementation of a computer literacy program for teachers:

All preservice teachers should be required to take a basic computer literacy course. Additionally, computer applications should become an integral part of teacher preparation programs.

A variety of inservice teacher education activities such as workshops, conferences and courses should be used by the schools for computer literacy training. States ought to approve computer literacy courses for recertification purposes.

Teachers of computing should be required to study computer science. Certification for this subject area is critical.

BIBLIOGRAPHY

Anderson, Ronald E., Klassen, Daniel L., "A Concepual Framework for Developing Computer Literacy Instruction", AEDS Journal, 1981, 14(3), 128-150.

Association for Computing Machinery, Computer Education For Elementary and Secondary Schools, 1981.

Beck, John J., Jr., "A Paradigm for Computer Literacy Training for Teachers", The Computing Teacher, 1981, 9(2), 27-28.

Board of Cooperative Educational Services, Third Supervisory District, Suffolk County, New York, "Developing Computer Literacy in K-12 Education", The Computing Teacher, 1981, 9(3), 43-48.

Dennis, J. Richard, "Training Preservice Teachers to Teach with Computers", AEDS Journal, 1978, 11(2), 25-30.

Milner, Stuart D., "Teaching Teachers About Computers: A Necessity for Education", Phi Delta Kappan, 1980, 61(8), 544-546.

Molnar, A.R., "The Next Crisis in Anerican Education: Computer Literacy", The Journal: Technological Horizons in Education, 5(4), 35-39.

Rawitsch, Don G., "Teaching Educators About Computing: A Different Ball Game", The Computing Teacher, 1981, 9(4), 27-32.

Introduction

The project addresses the instructional problems presented by students with handicaps which inhibit their ability to communicate. A major difficulty experienced by teachers of handicapped students is finding methods through which students may receive and respond to instruction, i.e. answer questions, practice skills which have been introduced, and demonstrate mastering of skills. The project uses systematic methods to identify and provide microcomputer-based devices and materials which will meet the needs of students with specific handicaps.

The program is based on a four-part rationale:

1) Adaptive devices are required to facilitate communication for orthopedically impaired students during instruction.
2) Microcomputers can be interfaced with adaptive devices to create a highly interactive learning environment for orthopedically impaired students.
3) Computer-based learnin methods are effective with, and motivational to all types of handicapped students, and can maximize teacher time.
4) The use of systematic methods can result in potentially useful instructional methods and products for all types of handicapped students.

The project is supplying handicapped students with a means of communication and with direct instructional support. Microcomputers are used to provide a communication link between students, teachers, and instructional materials. A three year plan is under way to develop and adapt a wide range of curriculum materials for use by handicapped students, including deaf, emotionally disturbed, learning disabled, mentally handicapped and orthopedically impaired students.

Software Selection

The initial effort was to find available software for the handicapped. Only three sources (Grover[1], MECC[2], and TRACE[3]) were identified. Grover supplies The Microcommunicator, which allows a student to select letters, words, and sentences from the keyboard. MECC supplies Special Needs-Volume I - Spelling, which presents three scanned choices of the spelling of a word in context, selection to be made by switch closure. The TRACE Center has pertinant publications in the field, and supplies the BLISSAPPLE program (Bliss symbols selectable by switch closure from a menu of nearly 2000 symbols.)

The Johns Hopkins First National Search for Applications of Personal Computing to Aid the Handicapped,[4] held in Washington, DC in November 1981, presented many excellent hardware and software developments which have since been used in this project. They are: I.O.R.[5], Alternative Communication System Project[6], Cope[7], Hillinger[8].

I.O.R. has developed a comprehensive group of programs for the physically handicapped, which allow program interaction through a variety of adaptive switches.

"The Evaluation System programs allow the evaluator to plug the user interface into the Apple-II microcomputer, and to give the user a step-by-step trial of the possible selection techniques. The programs offer a choice of pictorial, symbolic, and alphabetic "vocabularies", for both pre-literate and literate users. Their sophistication ranges from a choice of three simple line drawings to a choice of several hundred words. Using a language or symbolic system that is appropriate for the user, the evaluator can rapidly and easily test the user with all the different selection techniques, and determine the technique which is most appropriate to the user's abilities. In this way, the computer simulates the many types of augmentative communication devices which are potentially available to a handicapped user."[5]

"When the most appropriate interface, selection technique, and vocabulary have been determined using the Evaluation System, the user is ready for a communication device that he can use every day. The evaluator will have learned enough about the user's abilities to prescribe a commercially available electronic augmentative communication device. The Word-Com/Picture-Com programs for the Apple-II microcomputer are exceptionally versatile devices and should be considered for every candidate."[5]

"(Certain profoundly handicapped individuals may wish to be able to program a computer for educational, recreational, or vocational purposes. The non-verbal, severely motorically handicapped individual who can use some combination of a user interface and selection technique can use one of the "EZBAS" series of programs to totally control and program the Apple-II microcomputer.) Each of the "EZBAS" programs provides all necessary words and symbols on six "pages" on the video screen. As each word or symbol is chosen it is added to the program line on the bottom of the video display. When the carriage-return symbol is chosen, the program line is entered into the memory and the bottom of the screen is cleared for the next line."[5]

The Alternative Communication System Project offers adaptive switch control of Motor Training Games, Academics with Scanning programs, (Language Arts and Math) and Special Inputs program allows selection of two types of scanning, Morse Code, or assisted keyboard to be used in conjunction with many other programs designed for 'standard' keyboard entry.

William Cope has developed a machine language Morse Code program that allows any program to communicate with visually impaired students.

Michael Hillinger is developing a syntax driven, speech output program operated by a joystick. This program, "Augmentative Communication", uses formal rules of transformational grammar to enhance communication of the non-vocal.

An extensive survey of available Computer Assisted Instruction (CAI) programs was made, covering the educational needs of our students (ages 3-21) in Science, Mathematics, Language Arts, et cetera. Most programs require modification for the phyisically handicapped. This is an ongoing effort, to obtain modification approval from the vendors, make the modification (usually for single-switch operation), and provide for dissemination of availability.

Hardware Selection

Software selection determines hardware selection! The microcomputer was chosen on the basis of software availability. So many useful or potentially useful programs were written specifically for the Apple-II Plus (Apple Computer, Inc.) that it was chosen. The Amdek Color I monitor was chosen for its color rendition, resolution, and compatability.

Three speech synthesis systems were evaluated; tradeoffs had to be made in intelligibility and storage requirements. The Type'N Talk by Votrax was selected for its ease of use and unlimited vocabulary (through phoneme synthesis and text-to-speech translation system). A frequency control allows pitch variation; a 'female voice' synthesizer modification will be a welcome option when it becomes available. The monitor's amplifier/speaker is used for speech output.

One Apple disk drive with controller is also part of the system. The entire system is housed in a specially-designed console which allows placement of all equipment, single power switch control, and access by wheelchair users.

Interface Device Selection

Three sources have been identified for the interface devices: Prentke Romich Company, Zygo Industries, and I.O.R. All provide a wide range of adaptive switches which must be individually prescribed. Arm slot, eyebrow, foot, arm, and breath switches are available. Headstick use, for those students with controllable head movement, allows operation of the keyboard. A key guard is often used to prevent accidentally hitting adjacent keys.

An interface box is desirable to allow use of 'standard' 1/8" plugs to connect switches to the microcomputer.

Conclusions

New methods, software and hardware are becoming available and will be used or adapted as necessary. Some consultant programming will be contracted, where needs cannot be met by use of available software. New programming tools will be used: Apple Pilot[9], LOGO[10], BLOCKS[11], ELECTRIC POET[12] and High Resolution Secrets[13], each providing another means of program enhancement.

The second year of the project will address the needs of the mentally and emotionally handicapped at another school site in Sarasota. A 16-station microcomputer laboratory will be equipped to serve a large number of students.

The project currently has 10 working systems. Speech synthesis is used to provide communication for the non-vocal. CAI is being used by all students in the Physically Handicapped facility. Microcomputer usage has exceeded expectations.

- Warren Brown

Source Directory

1. Grover & Associates
7 Mt. Lassen Drive
Suite D116
San Rafael, CA 94903

2. MECC
2520 Broadway Dr.
St. Paul, MN 55113

3. TRACE Research and Development Center for the Severely Communicatively Handicapped
314 Waisman Center
1500 Highland Ave.
Madison, WI 53706

4. PROCEEDINGS OF THE JOHNS HOPKINS FIRST NATIONAL SEARCH FOR APPLICATIONS OF PERSONAL COMPUTING TO AID THE HANDICAPPED
IEEE Computer Society Publ. No.392
P.O. Box 80452, Worldway Postal Ctr
Los Angeles, CA 90080

5. I.O.R. Enterprises
Rt. 6, Box 20
Chapel Hill, NC 27514
Attn: Dr. Joseph T. Cohn, M.D.
Quotations used by permission from "The Word-Com/Picture-Com System" brochure.

6. Alternative Communications System Project WJ-10 CD 393
Child Development and Mental Retardation Center
University of Washington
Seattle, WA 98195
Attn: Judy MacDonald

7. Cope, William
Cannon Court Apts., I-5
Greenville, NC 27834

8. Hillinger, Michael
Vermont Department of Mental Health
Waterbury, VT 05676

9. Apple Computer Inc.
10260 Bandley Drive.
Cupertino, CA 95014

10. Terrapin, Inc.
678 Massachusetts Ave, #205
Cambridge, MA 02139

11. California School for the Deaf
39350 Gallaudet Drive
Freemont, CA 94538

12. Dr. Edmund Skellings
Florida International University
Miami, FL

13. Avant-Garde Creations
P.O. Box 30160
Eugene, OR 97403

Decentralizing School Administrative Support Services with Small Business Microcomputers in a Large School System. . .

Dennis L. Bybee, Ph.D.
Computer Education Coordinator
Department of Defense Dependents Schools

Schools need computer support services. No matter how large schools are or whether they are located in small towns or major cities, schools are increasingly required to manage mountains of paperwork on kids, their parents, their progress - or lack of it -, their use of entitlements, participation in special programs, attendance, and on, and on. The list of management information system requirements seems to grow even as school enrollments and financial resources are shrinking. Schools need computer support services more now than ever before.

Centralized school support services may not be adequate. School records and information management systems should exist for the benefit of kids. School personnel need to know more about kids than any other level in the educational hierarchy. While the state auditor wants to know how many students are in school throughout the year and even on a daily basis, the school administration needs to know where each student is every hour of the day, and the teacher needs to know what the the student is doing in every hour of the day. On a simple need to know basis, it should be clear that the largest amount of information is managed at school level. So, where do we usually find educational computers? In schools. Wrong. Most computers are in district, county, or state offices. Aren't they? Well they were - until now.

Decentralized school administrative support services are now thinkable for most - if not all - school districts. While school requirements demand - even beg - computer support, large equipment costs have been prohibitive and software has not been available for administrative support on the small inexpensive microcomputers that schools have increasingly been able to acquire for classroom use. The introduction of small business computers and management information system software into the marketplace now makes the unthinkable - thinkable; yes, even do-able, for as little as 8 to 20,000 dollars depending on school size and administrative requirements.

School districts, that want to take advantage of this technology, will need to identify required administrative support services, identify and select appropriate equipment, develop a personnel support network, and either acquire or develop appropriate software to accomplish the required support services. This is the approach that the Department of Defense Dependents School system has chosen to pursue in order to provide administrative support services to each school that it operates for dependents on military bases overseas. Schools have been grouped into school complexes according to the community feeder plan for elementary, middle, and high school attendance. There are 82 of these school complexes in 20 countries worldwide. Initially, each secondary school in the complex will have its own computer support; and each school will eventually have the capability of accomplishing its own management information requirements on in-house microcomputers.

Services identified for independent operation at the local school site include: (1) Assistance with STUDENT SERVICES TASKS such as registration, master schedule generation, scheduling, attendance, progress reports, and summary reports from the student services data base; (2) Assistance to other EDUCATION SUPPORT PERSONNEL in the utilization of computer technology to accomplish their program support functions; (3) Assistance in support of the automated MANAGEMENT OF SCHOOL RESOURCES; and (4) Assistance to school administrators in program PLANNING, EVALUATION, and IN-SERVICE training activities in the school.

Appropriate equipment includes three different configurations of the standard small business computer. The BASIC SYSTEM for schools or school complexes with enrollment greater than 300 students includes a central processing unit with at least 64K bytes of memory, two flexible disk drives, a CRT display/keyboard, a high speed printer, and system and application software. The UPGRADED SYSTEM for high schools with enrollment up to 500 students augments the basic system with an optical mark reader. The ENHANCED SYSTEM for secondary schools (including grades 7-12) with enrollment greater than 500 students augments the upgraded system with a 10 megabyte hard disk drive.

The personnel support network should include at least two key positions. There should be a school computer coordinator who can provide support services for a school or group of schools in a school complex and there should be a computer education coordinator available to provide overall coordination of administrative services for school complexes within larger geographic regions.

In addition to system software, the standard system should provide a student services application package and resource management software to include inventory functions, accounting functions, and electronic spreadsheet analysis functions. Word processing should be provided with system software. The software development plan should be to implement generic system and application software and to tailor programs to meet the specific needs of individual schools.

Implementation of this decentralized plan within the Department of Defense Dependents Schools will begin with prototype development in SY 1982-83 and will be extended to all 82 school complexes by SY 1984-85 in order to support administrative requirements of the 271 schools that are operated by the dependents school system overseas.

Readers wanting more information on this project should contact the author at the following address:

Department of Defense
Dependents Schools
2461 Eisenhower Avenue
Hoffman Building #1, Room 148
Alexandria, Virginia 22331
(202) 325-7810

Using Computer Graphics as a Critical Component in Mathematics Instruction

Janice L. Flake, Associate Professor
Florida State University

Diminishing mathematics scores on standardized tests, inadequate arithmetic skills, and an over-all lack of ability to operate in a functionally literate manner of too many students is a concern for many people. Considerable evidence shows major trouble areas are: fractions, decimals, percents, measurement, and problem solving.

One way mathematics can be viewed is as a long list of rules to learn and follow. Another way to think of mathematics is as a process of looking for patterns and relationships by building and using models. In the process of looking for such patterns and relationships, through exploring models, one can discover the rules and an over-all organizational structure.

The way one views mathematics learning affects the way one constructs mathematical learning experiences. With the computer movement, such a view will affect the way one uses computers and constructs computer materials. The intent of this paper is to explore this issue. In order to do so, research on some factors related to mathematical achievement, as well as factors involved in the process of mathematics learning, are discussed.

Some Factors Highly Correlated to Mathematics Achievement

An examination of the literature reveals factors highly correlated to mathematics achievement which include field independence, Piagetian logical operations, visual perceptions, ability to comprehend transformation of space, and problem-solving ability.

Field independence is the ability to focus upon detail and to detach one's attention from the surroundings. For example, one measurement of field independence is to be able to find simple figures embedded in a complex figure, that is, the ability to discriminate between the figures and focus upon specific characteristics. Field independence is highly correlated to mathematics achievement (1, 2, 18, 21, 22, 38, 40, 41, 45, 46). In addition, field dependent behavior (e.g., the inability to find simple figures embedded in complex figures) seemed to inhibit high attainment in mathematics (32).

A textbook page is often filled with much information. A student who does not know how to focus attention to specific information will often become overwhelmed. Many areas of mathematics involve the use of figures which sometimes become intricate; a student who is unable to focus his/her attention will have difficulty gaining the mathematical ideas from the figures.

The computer, on the other hand, can display materials piece by piece, so that the reader's attention can be focused. Instead of a whole page of problems being displayed at once, each problem can be displayed one by one. Drawings can be constructed, using careful timing, so that the student can give attention to appropriate parts. In addition, use can be made of different sound effects to focus one's attention.

Piagetian logical operations involve the way children think and reason about mathematical and logical concepts (27, 28). One central theme is the role of perception, that is information gained through one's senses (for example, seeing, hearing, touching), and how one thinks logically about the information. Included in this central theme is the concept of conservation; that is, physical and/or visual appearances are changed, but the mathematical essence remains constrant. Ability to perform conservation tasks is strongly related to mathematics achievement (4, 7, 13, 17, 25, 29, 30, 31, 32, 37, 39, 42, 44, 47). Both conservation of area (comparison of figures for which the areas remain the same but physical appearances change) (31, 39), and conservation of volume (comparison of volumes of quantities for which the volumes remain constant but physical appearances change) (31) have been found highly related to mathematics achievement. Many of these relationships show up more clearly in the intermediate and upper elementary levels of the elementary school.

Students who have not reached the level of coordinating visual stimuli with logical thinking should work with the physical manipulation of concrete materials until the consistency has become internalized. Computers hold the potential of being used in a very effective way, by guiding a child's action on concrete materials through transformation pictures of the concrete materials on the video screen, while the child follows with actual physical manipulation of the concrete materials.

Visual Perception is information received by the visual sense and interpreted by the brain, that is, the ability to understand and interpret meaningfully, information gained through the visual sense. Visual perception has been found to be highly related to mathematics achievement (12, 34, 35).

Tests can be set up on the computer to identify possible visual perception problems. The capabilities of the computer can also be used to vary the graphics to assist in developing the ability to perceive the visual materials. Such materials should be added to the set of diagnostic tests.

Ability to comprehend transformations of space has a strong relationship with mathematical achievement (9, 15, 21). Such transformations include rotations, translations, reflections, or a multitude of them. As these figures change, some people can adjust intellectually to the changes and relate the final and initial stages. Others can not adjust to those changes.

The computer can help guide the student in thinking about various transformations by transforming pictures on the video screen while the student manipulates physical materials if needed. Similarly, through the use of game paddles or identified arrows on the keyboard of the computer, the student can transform (such as, translate and/or rotate) pictures on the video screen.

Problem-solving ability is related to mathematical achievement (3, 21, 36). Problem solving is also highly correlated to spatial instruction (24), spatial visualization (11, 23), estimation (16), and Piagetian logical operations (25). The National Council of Teachers of Mathematics has identified problem solving as the most important aspect of mathematics learning that should be given explicit attention during the 1980's (26).

Computers can be very helpful in developing problem-solving skills in two major areas: (1) providing carefully prepared materials and (2) assisting the student in developing his/her own programs. One example of already prepared materials that can enhance problem-solving skills is the use of computer simulations (or working analogy of a realistic situation), such as a missile shooting a target. The student can assign values to the velocity and angle of incline for the missile projection. By changing these values, the student can begin to see patterns that govern the missile's behavior and can interact with the simulation to study the nature of that which is being simulated. A careful examination of such behavior should involve both analytic (taking apart) and synthetic (integrating skills).

In developing computer programs, certainly a student must be very analytical and logical; this is the basis for which the computer is designed. Furthermore, preliminary exploration suggests encouraging students to learn programming through graphics. Construction and use of computer graphics can be highly motivational for students, as well as help develop the students' spatial abilities.

Interrelationship among variables is currently being investigated by this author. Problem solving, ratio and proportion, spatial relations, logical inferences, and concrete/formal operations have been examined with several groups varying from fifth grade students to adults. Initial reading of the data shows that spatial relations is prerequisite for the other variables with the possible exception of logical inferences (10). Considerably more investigations will continue in this area. Note: ratio and proportion underlies the concepts of fraction, decimals, percents, and measurement. Hence, spatial relations appear to be prerequisite for the major trouble areas.

Some Factors Involved in the Process of Mathematical Development

The factors identified above relate strongly to mental imagery (pictures in the mind), and many show a strong relationship to the concept of space. Such abilities and concepts lend themselves to investigation via the graphic capabilities of modern microcomputers, where students interact with, and/or construct, such displays. In examining the process of learning mathematics, several factors need to be studied. Such factors include: the functioning of the brain, the role of models in mathematics learning, and spatial experiences.

Hemispheric brain research reveals that the brain is divided into two parts (the right hemisphere and the left hemisphere) that function in different ways. One very important aspect of brain functioning is: linear and sequential (left-hemisphere) versus simultaneous and holistic (right-hemisphere) learning. Another important aspect of differences in brain functioning is: verbal and explicit (left-hemisphere) versus metamorphic, visual, spatial, and tacit (right-hemisphere). Analytic, rational, and logical (left) versus synthetic and intuitive (right) form another important aspect of brain functioning. How does the whole relate to the parts and the parts to the whole? How does intuition relate to logic, and how does logic relate to intuition? Taking these ideas apart or putting them together is an important skill that should be developed; similarly, intuition and logic should relate to each other. Problem solving is, in fact, the process of integrating analytic and synthetic skills.

Many of the skills we wish students of mathematics to develop are: being able to organize their thoughts in a careful sequential pattern (such as following computation procedures or proving theorems); forming explicit definitions and ideas; and being analytical, rational, and logical. In other words, we are interested in developing left-hemisphere types of processes.

What about the right-hemisphere functions in mathematics learning? Many of the factors (visual perception, transformations of space, problem solving) discussed earlier relate to right-hemisphere activity. Furthermore, conservation tasks of the Piagetian-type perhaps test whether the visual-spatial notions (right-hemisphere activity) become coordinated with logical thinking (left-hemisphere activity) (43). Field-independence measures a coordination of analytic skills and visual perception and space; that is, a coordination between some right and some left hemisphere processing.

Much of our educational system has been centered around left-hemisphere processing. Yet considerable evidence suggests that we should pay far more attention to right-hemisphere processing. Several researchers in brain functioning (e.g., 8, 14, 20) have expressed serious concern that our educational systems have placed the bulk of their emphasis upon left-hemisphere activity. Hence, we may be educating only half (or perhaps much less) of the brain. Ignoring the right hemisphere may seriously lead to the stifling of one's mathematical development.

The role of models in mathematics learning is very important in the learning of mathematics since a model can represent an idea, relationship, operation, place or thing. A model can be static; that is, a fixed model to be looked at and/or studied. Similarly, a model can be dynamic; in other words, can be transformed from one situation to another, such as regrouping in arithmetic, transforming the areas of one geometric figure to another, or changing one's perspective in three-dimensional graphics. Perhaps the mystery of mathematics will become removed as more graphic representations become available to assist people to visualize various concepts and relationships.

Visual-spatial experiences influence spatial ability (5, 6, 19, 24). Students worked with various spatial construction activities, such as polyhedra model building and two- and three-dimensional puzzles making various configurations. In each case the students improved in their spatial abilities. If spatial abilities and mathematics achievement are strongly related, the inference then is that increased spatial experiences should ultimately increase students' mathematical performance. Similarly, experiences in space can be constructional via computer graphics.

Computers in the Mathematics Curriculum

Computers hold the potential to be used in traditional ways, through repetitive drill and practice and parceling out the rules or arithmetic to be followed, or through far more exciting ways of fully exploring the graphic capabilities to develop the visualization of concepts and relationships. Discussion of the mechanism of the brain is helpful in developing an awareness of the complexity of the learning/thinking processes. Much depends upon what is happening within the individual. Evidence already has been presented to show the importance of visual-spatial experiences. Further support of the importance of visualization is given by NCTM (26). The graphic capabilities hold much potential for developing such experiences. Programs can be designed to encourage spatial reorganization, visualization, and interrelationships of algorithm formation (for example, computation procedures) with visual experiences related to the physical world, which in turn can help coordinate right- and left-hemisphere experiences. The computer holds the potential for variability of stimuli; a sound, such as a beep or buzz, can be used at critical times to shift one's attention; one can add extraneous information, or keep the materials free of extraneous matter; motion can be used to illustrate spatial reorganization; and one can gain immediate feedback from the computer. Also students enjoy gaining control of the machine to develop their own graphics.

(1) - (47) See references.
Due to limited space here, the references are available upon request.
Contact:

Professor J. L. Flake
209 Education Building
Florida State University
Tallahassee, Florida 32306

COMPUTER-ASSISTED TEACHING OF ARABIC

J. Matthew Johnson and William W. Bonney Jr.

Research Associates, Interactive Research Associates, Iowa City, Iowa 52245

The aim of the following program is to teach the visual recognition of the Arabic consonants. (This is the first step in a "4-step" program for teaching the visual and aural recognition, and finally the written reproduction of the entire Arabic alphabet).

The opening display is designed to gain the learner's attention by focusing upon a location where it is useful to be able to read Arabic. A graphic of an airport and a sign written in Arabic provide additional "attention gaining" stimuli.

The objective of the lesson (to familiarize the student with the Arabic consonants) is presented in high resolution text immediately after the opening display. The term "familiarize" refers to an ability to visually recognize and identify these consonants in isolation.

In order to get the student to call upon relevant knowledge which he already possesses; he is asked to pick out the consonants from a set of English letters containing both consonants and vowels. This is relevant in two ways; he is distinguishing between consonants and vowels, and more importantly he is doing so in a manner and context which will be used later to make distinctions between various Arabic consonants.

The body of the lesson begins by asking the student to type the consonants in his own name using an English/Arabic chart above the keyboard. (They appear on the screen in Arabic letters). By looking at the chart, he is able to make any corrections which are necessary. This step involves the presentation of information (the Arabic consonants in the learner's name), learning guidance (the chart with the Arabic consonants and their English equivalents), performance (the student must type his name) and feedback (he must compare what's on the screen with what's on the chart).

After this the student is presented with a display of Arabic consonants. He is then asked to identify a common feature of the various consonants based on his observations. (If he recognizes the fact that they all have a "disconnected dot" he is given positive feedback and the program continues. If he fails to make this distinction the program loops back, flashes the dots, and repeats the question. This loop continues until the student answers correctly; at which time he is congratulated and the program continues.

The above task also involves the four events of instruction cited in the preceding paragraph. Information is presented (the consonants with one dot), guidance is provided (flashing dots), performance is elicited (the student must identify what the consonants have in common) and positive and negative feedback is also provided (the student is congratulated or his errors are identified and he is "looped back."

The next step in the program basically repeats the procedures and events of instruction described in the above two paragraphs. All the Arabic consonants are displayed and the student is asked to identify them based on the number of dots associated with each one (there are consonants with 1, 2, 3 and 0 dots). As he does so correctly the letters disappear from the screen. Mistakes prompt corrective feedback and the letter remains on the screen until it is properly identified.

After the student has caused all the letters to disappear from the screen by identifying them; he is congratulated and the program proceeds to the next phase. This last phase is designed to assess performance and facilitate transfer.

It does so through a game. The name of the game is "Quicksand Man". A graphic of a man whose name contains all the consonants in the alphabet (Rubinsklytromz Qwixgopjocvedefh) is presented. This man has just stepped into a quicksand bog. The learner must prevent the man from sinking into the quicksand. In order to do so he has to identify the consonants in the man's name as they randomly flash on the screen. Each consonant identified causes the man to move slightly upwards, while each mistake results in a downward movement. The game ends when the man sinks out of sight or completely rises to the top of the quicksand bog.

We believe the ability to save "the quicksand man" relates directly to the student's facility in recognizing the Arabic consonants. If he is unable to do so it is suggested that the entire program be reviewed.

The following question is relevant in determining how well this game asseses performance and facilitates transfer. Is it necessary to be able to recognize and identify the Arabic consonants in order to successfully play the game? While we firmly believe the answer is "yes, it is necessary"; the only way to be certain is by having a significant number of people work through the program and play the game. (A major reason for making this presentation is to facilitate this process).

In terms of "learning outcome" (Gagne, et. al.) the recognition and identification of the Arabic consonants is a "concrete concept". They are identified by "pointing and marking" rather than by definition.

One of the major requirements for learning to read, speak and/or write a language fluently is continuous interaction with a person (or persons) who speak it, over a relatively long period of time. What actually happens during this period cannot be completely duplicated by other means. However the extent to which this process is paralleled in the classroom or at the computer terminal relates directly to the effectiveness of any form of language instruction. This program was created with this principle constantly in mind.

It is also designed to be a prototype for computer-assisted instruction in the teaching and learning of all other non-Roman alphabet. It is felt that such a prototype is especially pertinent to the teaching and learning of languages such as Chinese, Japanese, Korean etc. which make extensive use of single alphabetic characters to represent semantic content.

END

TECHNIQUES FOR USING MICROCOMPUTERS TO AID LEARNING DISABLED AND MENTALLY RETARDED STUDENTS

Marguerite Mason
Coordinator, Project MASS
(Microcomputer Assistance for Special
Students), Title IV-C, ESEA
Charleston, Illinois

Walter Bruce Mason
Mason Computer Services
Mattoon, Illinois

The technology available through the advent of microcomputers can offer educators new opportunities for a multi-modal approach for teaching Learning Disabled (LD) and mildly mentally retarded (EMR) students. There exists a substantial population of these children in the United States; the 1981 child count needs data indicated over 140,000 EMR and LD in Illinois alone. Research conducted even before such features as sound, animation and color became available, indicated that computers can be an effective means of providing instruction.

Research such as that of Jamison, Suppes and Welles found that use of CAI led to savings in student time in learning tasks (Lance, 1977, p. 95). Several more recent studies reported by Forman (1982, p. 38), have confirmed this finding. The Jamison study also concluded that CAI used as a supplement to regular classroom instruction leads to an improvement in achievement, particularly for slower students. A review of research by a University of Iowa team concluded that "CAI, especially in the tutorial and drill and practice modes, is relatively more effective with low ability students."(Taylor, et al., 1974). In Forman's conclusions based on a review of more recent studies, she stated that "Tutorial and drill modes seem to be more effective for low ability students than for middle or high ability students;...Many reluctant learners become active and interested learners when involved in computer supported programs." (Forman 1982, p.38). Generally studies with handicapped students such as with EMR in New York (Nelen, 1972), and TMH in Calgory (Hallworth and Brebner, 1978, 1980) have shown increased academic achievement and improved self-concept for the experimental handicapped populations. Although these studies of handicapped students have taken place in segregated Special Education Classes, the implication has been that similar effects would be found in integrated, mainstreamed programs. These studies laid the foundation for the two projects which the authors of this paper have conducted in this area.

In 1979, the Apple Education Foundation (currently the Foundation of the Advancement of Computer-Aided Education) provided the authors with equipment to experiment in the development of CAI materials to attempt a multi-modal approach to teaching Learning Disabled students. This grant enabled the authors to review both existing CAI materials and Special Education theories to attempt to synthesize them into preliminary microcomputer programs for use with LD students. While this grant provided equipment and programming consultation, it had no provision for extensive field testing or controlled research. This is being done through Project MASS (Microcomputer Assistance for Special Students), a Title IV-C, ESEA funded program to extend the development work of the previous project and to test the effectiveness of CAI with LD and EMR students. Project MASS will provide two years of achievement data on over 150 LD and EMR students, K-12, who are using CAI and their non-CAI control group counterparts. In addition, it has been identifying modifications in design which can enhance the usability of software by LD and EMR students.

The general speculation is that the recent technological advances of color, sound, animation, voice synthesis, and other features available at a reasonable cost on microcomputers should enhance the learning process of handicapped students. In theory, this variety of stimulus techniques should enable the educator to select which channel(s) they would like to utilize in approaching the individual child; the student response may also be channeled through means such as keyboards, light pens, etc. There has been much speculation and some anecdotal records that CAI as so implemented on a microcomputer can be effective in meeting the unique needs of these handicapped students. Two of the major factors which have inhibited the use of this increasing available media by LD and EMR students are: the lack of good software identified as appropriate for use by handicapped students, and the lack of training for the educators working with these student in techniques for incorporating microcomputer assisted instruction into their instructional programs and in adapting existing materials to the unique needs of their handicapped students. Project MASS speaks to both these needs, but this paper will focus on some of the design features and techniques which can enhance the usability of software by this population.

Special Education students, like other students, generally have no fear of the computer. If they are not physically handicapped, most students quickly adapt to working on the keyboard even if they do not know how to type (Adaptive keyboards, joysticks, light pens and such ingenuous devices as grain elevator heat sensors which connect into the game I/O ports of an Apple are available to aid the physically handicapped). For the very young students, all but the needed keys should be disabled programatically, or, if that is not possible, masked by a piece of cardboard taped in place. Multiple choice answers involving numerals rather than letters should be employed wherever possible because they are easier to find on the keyboard and their use will reinforce numeric order. To encourage the use of the proper fingers on each key as more of the keyboard is used, colored sticky dots can be placed on the front face of each key to correspond with the finger to be used on that key. For example, the keys depressed by the right index finger (y,u,h,j,n,m,6 and 7) could be marked with a red dot. The student also has a red dot on the fingernail of his right index finger or wears a pair of rubber gloves with the fingernails painted the appropriate colors. Commercial typing programs such as "Mastertype" or, for older students, "Typing Tutor" can also be helpful in increasing typing skills. These programs will strengthen fine motor coordination, memory, and visual discrimination skills as well.

Programs should be kept as jargon-free as possible. However, the student should be taught such common words as return, space bar, select, and menu. The students must be able to control their own reading rate except when increasing their reading speed is the objective. Consistency is important; e.g., the student shouldn't press the space bar to move from one to another sometimes and the return key other times.

Directions should be clear and concise. All too often programs employ 5x7 uppercase only dot matrix characters displayed single spaced in a manner which makes the words difficult for students, particularly ones with visual disabilities, to read. For best effect, the amount of text displayed at one time should be limited, keeping in mind the effects of pleasing visual appearance versus clutter. However, it is a myth that displays must be plain to be effective with Special Education students. Some students may tend to be easily distractable and tend to focus on something other than what is being taught. The use of such stimuli as color cuing, flashing, underlining, arrows, boxes or animation to highlight the important aspects can focus the students attention, but should be used with judgment and moderation. Razz-a-ma-tazz with no purpose accomplishes nothing, but students can handle something as elaborate as an arcade type format if it is well designed with a particular objective in mind. Also they are a lot more apt to stick with something that is interesting.

One of the reported positive effects of CAI which generalizes to other situations is that students tend to read and follow directions more attentively. Often commercial programs add extra verbage which only frustrates students who have difficulty reading them. For example, one of the more educationally useful programs availabale from Apple is "The Shell Games", which provides the framework for the teacher to put in their own true-false, multiple choice, or match column questions for the students to drill interactively. The title page of the program proclaims it to be

```
        THE SHELL GAMES

      BY BRUCE TOGNAZZINI

   COPYRIGHT 1979 BY APPLE INC
```

followed by a menu of available programs. After the student makes his choice, he is presented with a frame like this:

```
  &&&&&&&   MR. MULTIPLE   &&&&&&&

        BY BRUCE TOGNAZZINI

  COPYRIGHT 1979 APPLE COMPUTER INC

            PRESENTS....

LITTLE KNOWN AND LONG FORGOTTEN FACTS

             A QUIZ BY

            MR. MULTIPLE

  COPYRIGHT 1979 APPLE COMPUTER INC
```

Then the student is asked to type in his name. We stress to the student that he must read carefully every word in the directions, yet he is shown three copyright notices within a few seconds. One can also imagine what encountering the name "Tognazzini" can do to a student who has reading difficulties. However, Mr. Tognazzini has graciously given his permission to imbed his name in a remark statement within the program rather than it appearing on the screen. The program is so well documented that simplifying the directions is an easy task.

"The Shell Games" can be a very effective program for use with Special Education students. For example, all high school students in Illinois are required to pass a test on the U.S. Constitution prior to graduation. This fifty question test has generally proven to be a stumbling block for our EMR population. In previous years many of the students have required multiple attempts to pass and some have not passed at all. This year multiple choice and true-false drills on the U.S. Constitution were put on the microcomputer by the teacher who did not know what the actual test questions would be. Each student drilled on the microcomputer for 30-90 minutes total during the two weeks prior to the actual test. Every single high school EMR student who took the U.S. Constitution Test passed it. Not only that--every student passed it on their first attempt, and two of the students answered 49 of 50 questions correctly! The average score of the EMR students was higher than that of the general school population on the test.

Two of the features of this program which may have contributed to its success are its format and its adaptability. The students worked on the material in a non-threatening, almost game-like format which provided immediate reinforcement for their success and remediation for their errors. Their mistakes could not embarass them in front of their peers, and they could work at their own pace. In fact, many of the students didn't seem to realize that they were working. The questions and explanations were written in vocabulary that they understood. When a reading level analysis is performed on many programs, they are found to actually be at a much higher reading level than they claim to be. Even using the right reading level won't insure success. A high school student who is reading at fourth grade level may be insulted by the gimmicks and tone used in a program designed for a fourth grader.

A good program is characterized by its flexibility and adaptability. It should anticipate student errors like entering letters where numbers are requested and provide error trapping for them. It must be "user-friendly" and never put a student down with statements like "You Dummy!" when he makes a mistake. On the other hand, making errors should not be more fun than getting the correct answers like in some hangman type games. The student should always have a way to end the program at any time such as pressing the ESCape key. Wherever possible, the computer should automatically keep records on the student's errors so that the teacher may use them for later diagnosis and prescription. The teacher should have the option of removing the sound, altering the number of problems to be done, the time allowed for speeded drills, etc., in order to tailor the program to the student's individual needs. Whenever possible, several different versions of the same program should be developed. Each should highlight an approach that exploits a different learning channel. Students with learning problems often have difficulty transferring and generalizing learning. Care should be taken to encourage and facilitate this transfer. In many cases follow-up worksheets can be of assistance.

These points are just a few of those which should be considered when evaluating or designing software for use with LD or EMR students. The use of additional peripherals such as light pens, tape recorders, and synthesizers can further enhance the effectiveness of some of the software. Properly designed software can provide high interest, self-paced materials which can incorporate repetition and remediation where needed. This software, in the hands of properly trained and motivated teachers, can make microcomputers effective tools in teaching our Learning Disabled and Educable Mentally Retarded students.

REFERENCES

Forman, Denyse, "Search of the Literature", The Computing Teacher, Vol. 9, No. 5, January, 1982; pp. 37-51.

Hallworth, H.L. and Brebner, Ann, "Computer Assisted Learning for the Mentally Retarded", Calculators/Computers, Vol. 2, No. 4; April 1978; pp. 78-80.

Hallworth, H. L. and Brebner, Ann, "A Multi-Media CAI Terminal Based upon a Microprocessor with Applications for the Handicapped", Paper presented at the Annual Convention of AEDS, St. Louis, Missouri, April, 1980.

Lance, W.D., "Technology and Media for Exceptional Learners: Looking Ahead", Exceptional Children, Vol. 44, No. 2, October, 1977; pp. 92-97.

Nelen, Edwina, An Evaluation of Computer Vocabulary Instruction with Mentally Retarded Children. Research Report No 7322, Syracuse City School District, New York, August, 1972.

Taylor, S., Edwards, J., Norton, S., Van Dusseldorp, R., and Weiss, M., "The Effectiveness of CAI", Paper presented at the Annual Convention of AEDS, New York, New York, May, 1974.

COMPUTER LITERACY - TEACHER EXPECTATIONS AND STUDENT ACHIEVEMENTS

Jack McCabe

Recently I was called on to design a computer literacy course for high school students. The only prerequisite for the course was to be Algebra 1. Having been a mathematics teacher for the college bound I approached my task with anxiety. Here is just a partial list of my concerns:

Is learning with a computer also learning about a computer?
How could I make this learning an active experience?
Could all students learn to write programs?
How could I ask students to exhibit an awareness of computers?

Because computer literacy is an emerging curriculum I could not simply copy another teacher's outline. Instead I would use materials picked up at workshops, borrow ideas from magazine articles, and include examples from my experience in data processing and in mathematics.

I decided on the one semester approach for two reasons. First, was my concern with having to deal with students who could not benefit from the academic concepts a longer course should include. Secondly, this schedule would allow me to reach twice the number of students in a year.

I settled on the following course description and objectives:

> Computer mathematics is a combination programming and computer awareness course which attempts to give each student extensive hands-on experience with problem solving using the BASIC language. In addition to homework and laboratory exercises, each student reports on the impact of computer technology in our society.

Specifically each student will:

Trace a procedure given either as a sequence of steps, or as a flow chart, or as a BASIC program.
Translate procedures into programs.
Write routines to calculate a summation, to perform a search, and to manipulate a list.
Create pictures using the graphics capabilities of the language.
Manipulate alpha-numeric expressions using string functions.
Write short reports based on video material, popular publications, and discussions. Topics include - The threat of computers on employment and privacy, - Computers can predict futures, aid in design, and control environments.

Armed with these plans, eight microcomputers, and solid support from the administration I took on some sixty students in two classes. They came with a variety of reading and math skills. Their study habits ranged from the best to the worst. But I knew early in the semester that I had found a new calling. At the end they were begging for a follow up course. Every child was able to write a program provided the procedure was logically simple. I could detect no relationship between previous math achievement and the ability to create a program.

The students enjoyed the freedom and social atmosphere which accompanied the laboratory sessions. Most enjoyed the discussions that followed the video material. Many took to the special challenge that program creation offers. Some were successful on examinations; tests requiring the lonely recall and logical discovery of answers and methods seen somewhere before. A few failed.

I made a special effort to relate the awareness topics to the laboratory activities. A program to code and decode secret messages was assigned following the awareness topic of privacy and military security. Students translated sheet music into computer code which was then played through an amplifier. We watched computer generated moving pictures created when some students modified a graphics program. They experienced art from robots, intelligence from artificial beings.

I will offer a second course to focus on computer science and more programming, but it will be open only to those students judged successful with the demands of the logic. As for my future in pure mathematics, or the school's willingness to develop new curriculum, may I sing you that song, "How are you going to keep them down on the farm...?"

COMPUTER LITERACY FOR EDUCATIONAL ADMINISTRATORS

Introduction

The importance of the computer in society today does not have to be demonstrated. With the advent of the personal computer, we are being confronted by this technology almost everywhere we turn. Reading the popular literature, one might perceive that the computer can significantly improve the efficiency of any activity with which it is associated. Such a perception is hard to reconcile with the many "horror" stories to the contrary. A more accurate portrayal would result in a perception that under the control of a literate user, the computer has the potential for significantly improving the efficiency of any activity with which it is associated.

Computer literacy is the key to the effective integration of the computer into any activity and today we find computer literacy courses being offered by every institution where there is a need for computer use. There are computer literacy courses for teachers, engineers, doctors, etc. A particular group that has not had the benefit of tailor-made courses is educational administrators. This is particularly unfortunate since the continued growth of literacy courses, as well as the integration of computers into the curriculum, depends upon decision makers in colleges and universities being convinced of the importance of the use of computers. In these difficult financial times, only the computer literate administrator recognizes the importance of adding new programs in computer literacy, justifying the purchase of additional hardware, and hiring of additional staff to support the same. Furthermore, it is the computer literate administrator who will also be able to integrate the computer into his or her institution's administrative practices in order to save the necessary time and money to justify the implementation of new programs.

In this paper, we describe a strategy being used at Cuyahoga Community College (CCC) in Cleveland, Ohio, for helping college administrators to become computer literate. Each quarter at CCC, we distribute a minimally configured personal computer with communications capability to each of approximately fifteen administrative-level employees. The administrators can use the personal computer at home for the quarter (about ten weeks) in exchange for attending a seminar, once a week, for two hours of instruction. In the seminar, they learn how to use the computer as an administrative tool in the college. At home, they not only have the opportunity to practice the various applications introduced during the seminars but also, get to "play" with the computer with their families. After each quarter, the seminar participants return the computers and the project is replicated with a new group of administrators. The first group meanwhile begins to make greater use of facilities available to them within the college.

The project, now in its third quarter of implementation, has shown not only that all the seminar participants have been able to overcome their anxieties about computers in general, but also that they have even become enthusiastic about using computers in conjunction with their administrative duties. As a result of the seminars, college administrators have begun to communicate using electronic mail, to incorporate word processing in their written communications, to use the

309

college's on-line student information system and financial management system creatively in managing their units and solving their problems, and, to use computerized modeling techniques for budget planning and resource management.

The Administrative Literacy Project

The Administrative Literacy Project addresses two issues: the anxiety that administrators (and many other adults) have for computer technology in general and its potential impact on their lives, and the spectrum of applications available to literate administrators to help them perform their assignments within the college more effectively. Our objective is both to relieve administrators of their anxiety and to have each individual work through several of the many applications available.

Our experience suggests that the best way to overcome computer anxiety is to provide an environment within which administrators have convenient access to the hardware. By making available a minimally configured personal computer that administrators can take home and "experiment" with, the college has provided an invaluable means for helping administrators overcome computer anxiety. Using an ATARI 400 connected to television sets at home, the administrators can use the computer for writing and running small programs in a high level computing language, for executing simple, cassette-based applications that demonstrate more advanced computing concepts, for using the various systems available through the college's computer, and for playing games. Many of the participants in the literacy course have indicated an interest in purchasing their own personal computers.

The administrators attend a weekly seminar at which they are introduced to the spectrum of computer applications available through the college's computer system. To facilitate their experimentation with these applications, each administrator is given not only an Atari 400 (with 16K of memory, a basic cartridge, and a cassette recorder), but also an interface unit (Atari 850), an acoustic coupler (Atari 810), and a telelink cartridge so that he or she can communicate with the college's Honeywell installation.

Each week, a new topic is introduced for "practice" during the remainder of the week. Applications that are of particular interest can be pursued in greater depth with the help of computer center staff at the conclusion of the project. Although participants relinquish their equipment at the end of ten weeks, most, if not all, have ready access to terminals in or near their offices (a fact that many were unaware of before they participated in the project).

Although the topics covered in the project seminars are discussed in highly specific terms relating the concepts to CCC's hardware facility, the following description of the topics is general and highlights the importance of each to the college administrator.

Week 1 - General introduction and distribution of the personal computers. This first session at which we demonstrate the personal computers ("boxes" a La Spero) and how to connect them is like "Christmas or Chanukah in July." Many administrators approach their boxes as they would a strange and mysterious toy. They are excited and highly motivated to get started. The administrators are given a telephone number to call in case they have trouble connecting their computers at home.

Week 2 - Logging onto the Honeywell and electronic mail. The

310

most formidable task in all of computer literacy for most people is getting logged onto the "mainframe." Establishing user ID's and passwords and working through the sign-on protocol has proven to be the most difficult part of the entire literacy project. It is clear now that not being able to "log-on" has been the greatest barrier to effective use of our facility. As a first application, we demonstrate how to use the electronic mail system available on our Honeywell. This system is rapidly proving to be one of the most popular and practical applications of the computer in the College. It is facilitating communication among administrators as no other technology (including the telephone and the photocopy machine) has done before. It is now possible for administrators to communicate at any time of the day or night by leaving messages, sending memoranda to individuals or entire groups, making announcements, soliciting information, etc., conveniently and quickly.

Week 3 - Electronic mail continued and introduction to the concept of a computerized file. Ideas on how to use mail effectively as a tool of administration flow freely the week after it has been introduced. Many of the ideas include sending messages to small groups within the college (e.g., all of the deans of instruction, members of the provost's cabinet, etc.). This leads very naturally to the concept of a computerized file. The participants learn how to create a file using a text editor called Fred (friendly editor). They learn how to create a file, enter information into it (the names of user ID's to whom they want to send mail, how to save it, and how to use it actively in conjunction with mail).

Week 4 - The use of Fred continued and introduction to the student information system (SIS). Another application of files and Fred is preparing messages ahead of time and editing them before distribution. This can be accomplished using Fred after learning a few more commands. During week 4, we also introduce the college's on-line student information system. This very powerful data base system permits the administrator to access up-to-date information about students and courses. The information includes both student records and enrollment data. While many of the administrators may have been forced to use data from the SIS before (primarily through computer printouts), they are now ready to appreciate its potential.

Week 5 - SIS and Fred revisited. With this introduction to file management systems many of the administrators begin to inquire about the possibility of setting up their own file management systems using data from within their departments. At this point, we look again at Fred and its potential for formatting computerized record keeping systems. Although we use the Honeywell at this time, we begin to mention the potential of personal computer technology for addressing departmental needs of administrators. From here on, we demonstrate the potential use of a personal computer (an Apple II 48K, dual disk drive system) in conjunction with the applications that we present.

Week 6 - Text and word processing. We now add the use of Roff, a text processing system used in conjunction with Fred. This text processor on the Honeywell is a primitive word processing system that the participants can use to get a "feeling" for the potential word processing has for assisting them in their tasks. Although many of the participants have had access to word processing equipment, they now have a greater appreciation for its potential.

Week 7 - Pause and regroup. At this point in the project, we allow time for the administrators to catch up, digest all the material that has been presented, and to ask questions and review.

Weeks 8 and 9 - Financial Planning Systems. The brief respite of week 7 is needed because this last topic is much more complex and very important. The last major topic in the project is a discussion of the use of the computer for financial planning within the college. Using IFPS (the Interactive Financial Planning System from Execucom), a very versatile and flexible financial planning and modeling system available on the Honeywell, the college has implemented a number of models that administrators can use in planning their budgets. Since IFPS is, in fact, a computer language, we must spend more time introducing it and its applications. We very carefully develop a series of simple exercises the first week and then explain more complex models in the week that follows.

Week 10 - Summary and future directions. During the last session we demonstrate Visicalc as a tool for modeling on the personal computer and conclude with a discussion of future directions. By this time, virtually every participant is clamoring for his or her own terminal and possibly even a personal computer on which to do word processing, construct data files, and carry out financial planning and budgeting. There are also requests for communications capability into the Honeywell so that they can use existing applications (mail, SIS, IFPS, etc.).

Conclusion

The changes that take place during the ten weeks of the project are dramatic. Instead of indifference, we now have enthusiasm; instead of anxiety, we now have frustration ("when can I get my own CRT")? These emotions are a good sign for the future because we will shortly be integrating personal computer technology with large mainframe technology. This will provide maximum benefits to the college, its staff, and its students. The "key" appears to be a literate administration; making technology available for them to use at home appears to be the "box" in which the key can be found. In the words of one of the computer center staff who now has to deal with literate administrators who are clamoring for facilities, courses, and software packages, "you've opened a box alright, a Pandora's Box. But then, that is what we have been waiting for. . ."

- Samuel Spero
Jon Jonas

312

EVALUATION OF THE IMPLEMENTATION AND MANAGEMENT OF MICROCOMPUTERS FOR INSTRUCTION

Ralph Van Dusseldorp
Professor of Education
University of Alaska
3221 Providence Drive
Anchorage, AK 99508

In the past considerable attention has been devoted to evaluation of the effectiveness of computers for instruction. The typical method of studying the effectiveness of computer assisted instruction (CAI) has been either by teaching one group of students by computer and another group by conventional methods and then comparing the results, or by using CAI to supplement conventional instruction for a group of students and comparing the achievement of those students with that of students who received traditional instruction only. In general, those studies have shown that the computer can be a very effective instructional tool. However, the potential of CAI is not always realized. In a number of situations microcomputers have been provided for teachers but the teachers have not used them effectively, if at all. In some schools and districts microcomputers are being used extremely well while in others their use is quite ineffective. We are discovering that it may not be enough to just purchase a microcomputer and place it in a school in order to have it well used.

Prior to microcomputers, when the acquisition of a computer for instruction was a major expense, schools and school districts went to considerable effort in planning for the implementation and management of CAI. When the expenditure for a computer was large, it made sense to plan carefully and to provide a budget for necessary components in addition to the computer itself. Now however, with the purchase of inexpensive microcomputers we are likely to do little planning for implementation and management and to provide little if any budget for related components.

Even though the cost of a single microcomputer may be small, a school or district may end up purchasing a quantity so that the total cost is large. Even though the total cost is not large, the computer can be an important instructional tool and it behooves a school or school district to carefully plan for the implementation and management of CAI. The purpose of this paper is to provide a model for evaluation of the implementation and management of microcomputers for instruction. It is hoped that this model will be useful for evaluating systems that are already operating as well as in planning for new systems. The model presented is based on a variety of sources. It is based on principles of sound management, and successful practices in the implementation and management of other educational innovations, on experience with CAI in a variety of schools and school districts, and an intensive study of the implementation and management of microcomputers for instruction in one medium sized school district -Anchorage, Alaska. Anchorage has about 50 elementary and 12 secondary schools. It has at least one microcomputer in each school and as many as eight in some for a total of over 100.

The model for evaluation of the implementation and management of microcomputers for instruction is presented as a series of evaluation questions, with each question followed by a discussion of the factors involved.

1.0 **Is there a clear purpose for the installation of microcomputers?**

There are a variety of possible purposes for installing microcomputers for instruction. They include, among others, to supplement instruction for slow learners, to provide enrichment for the gifted and talented, to provide drill in the basic skills, to teach computer science, and to provide computer literacy. There may be a number of purposes. Whether there is one or many, they should be clearly stated.

1.1 **Is the purpose known to teachers and administrators?**

Often central administration and the school board are well aware of the purpose microcomputers are purchased to achieve, but other administrators and teachers are not made aware of the purpose. All affected parties should be aware of the purpose.

2.0 **Is the computer an integral part of the curriculum?**

Microcomputers should be a planned, integral part of the curriculum. When appropriate, they should be included in the district curriculum scope & sequence.

3.0 **Is someone in charge of microcomputer implementation and management?**

There are a number of possible choices for placement of responsibility for microcomputers. Typical choices include the mathematics department, the business education department, the District business department, the District data processing department and the media department. Often the math department is responsible because math teachers are the initial users, or the district data processing department is responsible since that department already knows about computers. Responsibility should be placed in the area that will most likely provide effective management for instruction. In many schools and districts the most appropriate area will be the media department. That department is likely to treat the microcomputer for what it is - a part of instructional media and can provide unbiased service to all areas. It is difficult to convince teachers in other subject areas that the math department will provide them equal access to microcomputers.

4.0 **Is adequate budget provided for necessary components in addition to purchase of hardware?**

All too often money is provided for purchase of a microcomputer and nothing else. In order for the hardware to be used effectively, money must also be provided for software, supplies, maintenance, inservice training and management.

5.0 **Are the microcomputers throughout the district compatible?**

A variety of brands of microcomputers are currently available. They are not compatible with each other. For a variety of reasons - including cost, maintenance, inservice education, transportability of software and convenience - all or most microcomputers in a district should be from the same manushould be from the same manufacturer. Left to make their own decisions various schools and departments will choose different microcomputers. Each will feel that their choice is best. There are no bad microcomputers on the market today nor, as of this writing, is there one that is demonstratively better than the others. Compatibility of microcomputers in a district is much more important than the particular brand chosen.

6.0 **Are the microcomputers located in the proper places to achieve their purposes?**
It is not at all uncommon to find microcomputers purchased for particular purposes to be located in places that make the fulfillment of those purposes impossible. For example, if the primary purpose of microcomputers in a district is to provide drill and practice in basic skills for slow learners, that purpose is not likely to be met if the microcomputers are to be found in administrators' offices, teachers' work rooms and lounges, or mathematics classrooms. Just as there are specific advantages to having the media department responsible for microcomputers, there are advantages in placing microcomputers in media centers or libraries where they are accessible to a variety of students.

7.0 **Is there congruence between the stated purposes of microcomputers and the actual use?**
Often microcomputers are purchased for one purpose and end up being used for quite a different purpose. Commonly, for example, they are purchased to provide CAI but are used in the principal's office to keep track of attendance or used by teachers to record and compute grades, or they are purchased to provide enrichment for all students but are used only in the math department, or they are purchased for instruction but are used primarily for game playing. That does not mean that the alternate uses are not legitimate or that microcomputers should be used only for the original purpose. However, if the original purpose is still valid, then the alternative uses should be examined to see that they do not distract from the original purpose. Of, if the original purpose is not longer valid, it should be changed.

8.0 **Is appropriate inservice education for teachers and administrators provided?**
For effective use of microcomputers for instruction, inservice education is essential. Without a planned program of inservice education there is likely to be little use of microcomputers beyond that of a few particularly motivated teachers. In most cases it is not enought to provide training on how to operate microcomputers and how to write programs. Rather, teachers need to be taught to use microcomputers as effective instructional tools. In the future, as more teachers become knowledgeable about microcomputers through their preservice training, extensive inservice training will not be necessary. Now, however, few teachers have received preservice training in the use of microcomputers. Thus inservice training is

needed.

9.0 **Is maintenance provided for?** Microcomputers break down periodically and need repair. Repair can be time consuming and expensive. Provision should be made for timely maintenance and repair with as little loss of instructional use as possible.

10.0 **Is appropriate software available to teachers?** It is easy to obtain a microcomputer. It is difficult to obtain appropriate software for many instructional purposes. Software is available from a wide variety of sources, both public and commercial. The software serves a multitude of purposes and varies greatly in quality. The task of locating available software, then choosing the appropriate from the available is horrendous. But if appropriate software is to be made available to teachers, a considerable effort must be made to search for, screen and obtain software.

10.1 **Is the software available readily accessible to teachers in a usable form?** Even though a district or school makes a wide variety of microcomputer software available to teachers, too often the teacher who wishes to select computer programs for instructional use is faced with a very difficult task in choosing software appropriate for his/her use. It is not uncommon for a teacher wishing to choose programs to be faced with little other than a group of disks containing dozens or even hundreds of programs. The teacher must sort through many to find a few that might be appropriate, then must intensely study (perhaps run and rerun) those few to find the best. Many, if not most, teachers faced with that kind of situation will soon give up trying to find appropriate programs. Computer programs should be cataloged and indexed in much the same way that films are. The index should include at least grade level, subject area, topic and type of instruction. If possible it should also include some evaluative comments about each program. A teacher should be able to go to the index/catalog and quickly identify the programs that fit the general categories of his/her needs. The teacher should then be able to quickly select those programs and review them for final selection.

FULFILL INSTRUCTIONAL COMPUTING NEEDS WITH VARIOUS MICROCOMPUTERS

Dr. Kenneth E. Brumbaugh
Minnesota Educational Computing Consortium
2520 Broadway Drive
St. Paul, MN 55113

This paper and the associated presentation will focus on implementing microcomputer hardware into instructional computing settings. A quick review of representative microcomputers from three different price ranges, less than $500, $500. - $2500., and above $2500., will be contrasted. Particular emphasis will be placed on which computer literacy activities could be provided at the different cost levels. An analysis of initial and continuing costs will also be made for each of the three price ranges. Finally an overview of low-cost microcomputer networking will be presented for the purpose of informing the participants as to how to save money while obtaining additional microcomputer stations.

More Americans should become computer literate if we as a nation are to maximize our problem solving abilities and be more productive in our daily lives. The personal computer is in our lives and will be even more so in the future. Have you notice that personal computers can:

- enable individuals to save energy resources and costs
- allow governments to operate by handling the necessary data
- enable individuals to travel more efficiently and effectively
- enable doctors to verify all prescriptions for medication
- enable individuals to do all banking tasks without travel
- allow commercial establishments to lower costs
- allow offices to improve the quality of work and life
- allow individuals to have education, entertainment, and a utility all in one device.

What about instructional uses of personal computers, both in schools and in the home? Will there be a computer in every classroom? The answers to questions such as these will influence the extent and method by which computer literacy programs are incorporated into schools and homes. As society makes more use of personal computers, so will schools. Thus the development of a school curriculum which includes a computer literacy component is essential.

What is meant by the term computer literacy? Most individuals, including those who are in the computer field, are not comfortable and confident of their definition of computer literacy. There is not one right definition, but it is important that each person forms a definition of this intangible concept. A definition might be as follows. To be computer literate one must be able to define, demonstrate, and/or discuss:

- how computers are used
- how computers do their work
- how computers are programmed
- how to use a computer
- how computers affect our society

Perhaps the biggest issue is whether computer programming should be a requirement in computer literacy curricula. If computer programming is not a requirement, then hands-on experience using a simple, relatively inexpensive, portable, non-violent piece of equipment called a personal computer most certainly is.

Computer literacy components designed for instructional purposes should have as their primary focus individualization of instruction. The content could include training in particular skills, financial assistance, work assistance, entertainment, utility, and hobbies. If individuals are prepared for the advancing technology, they will be better able to understand how power, skill, control, intelligence, and policy can be influenced and used properly.

Individualized materials are becoming available and allow worthwhile instruction to take place, whether it be in the home or the school. If computing applications result which bridge computing and individualized instruction, then better educated and more computer literate citizens will result. But, before any of these goals can be realized, an adequate number of instructional computing stations must be available for student use. What type of instructional computing services should you chose? A look at timesharing vs. microcomputing, then a look at various microcomputer options, should help answer the question.

Before making the decision whether you should obtain timeshare service to fulfill your computing needs or acquire personal computers to do the job, it is essential that you prepare a list of the usage characteristics, or modes of usage for which you intend to use the computer. If you choose ease of use, reliability, responsiveness, high student access time, portability, feedback, graphics, and sound as the characteristics of your use, then you should most certainly acquire a personal computer to do the work for you. If printed output, large data sources, high amounts of student record keeping, and multiple languages are the necessary ingredients in your intended use, then perhaps you should consider obtaining, or continuing, your timeshare service. If you decide to select the type of computing service based on your essential modes of usage, then you might follow this suggestion. Computing uses such as drill, problem solving, programming, simulation, and tutorial applications lend themselves best to the personal computer. When you do make this decision, be sure to consider all cost factors, such as purchase, lease, installation, maintenance, telephone, and operations.

Figure 1 Possible Instructional Computing Budget - Large District displays some of the possible budget items and amounts which an active instructional computing district might realize. Notice that the cost of the initial computing equipment is projected to be only one-half of the total budget necessary to support the instructional computing program.

Item	Budget Amount
Salary & Benefits	
Full-time	40,000
Part-time	6,000
Rent/Repair	
Space	2,000
Equipment	5,000
Printing	1,000
Honorariums	
Internal Consultants	500
External Consultants	2,500
Communications	
Postage	200
Telephone	500
Travel	
In-State	2,500
Out-State	2,500
Fees	
Tuition	250
Memberships	250
Supplies	
Office	300
Computer	1,000
Software	
Purchase	2,500
Licenses	7,500
Equipment	
Furniture	1,000
Computers	75,000
	$150,000

Figure 1. Possible Instructional Computing Budget - Large District

Two additional cost questions are: 1) How expensive/complete does the equipment have to be?, and 2) How long are you expecting to use the equipment? Perhaps the information given in Fig. 2 Computer Literacy: What, Where, When would be helpful in determining the type of computer equipment to obtain for your school's computing program.

It is necessary to determine what your goals are, where you intend to implement computing into the curriculum, for how many students, and with what software prior to selecting the price level of computing hardware needed. The information contained in Fig. 2 should only be used as a guide and verified, or replaced with what is appropriate for the school in question. The point is that various priced computing equipment will serve various needs.

Computer Literacy Goal	Price Range(1) Of Needed Computer	Curricula Fit:(2) Level of Instruction Elem	Jr.High	Sr. High	Percent of Students Affected	Software(3) Needed
How to use a computer	L - M - H*	R	R	R	100	M
How to program a computer	L - M - H	P	R	R	33-66	L
How are computers used	M - H	I	P-R	R	50-75	H
How computers impact society	M - H	I	P	R	100	M
The science behind computers	M	I	P	P-R	5-10	L

(1) L = under $1000.
M = $2500.
H = over $5000.

(2) R = recommend placement
P = possible with right circumstances
I = inappropriate/unnecessary

(3) L = low
M = medium
H = high

* Any symbols separated by a dash means either (any) could meet this need.

Figure 2. Computer Literacy: What / Where / When

	Cost Per Configuration*		
Item	Low	Medium	High
Original Equipment	$1000	$2500	$5000
Maintenance/Repair	$ 500	750	1500
Software/Documentation	$ 500	1000	2000
Supplies	$ 250	250	500
Five Year Total Cost	$2250.	4500.	9000.
Five Year Cost/Hour/Station	$ .5	1.	2.
Cost/First Year	$1400.	3200.	6400.
Average Cost/Other Than First Year	$ 213.	325.	650.

*Assumes:
1) A five year usage period
2) 180 school days/year + 5 hours/school day
3) All costs are approximations

Figure 3. Instructional Computing Cost/Station

Furthermore, various priced equipment generally will mean that the on-going costs related to using that equipment will be somewhat proportional to the initial cost. Fig. 3. Instructional Computing Cost/Station shows one estimate of average five-year costs associated with three different priced equipment configurations.

If after reviewing needs, goals, and equipment offering, you find that all factors are even, then you should select the least expensive equipment because that provides the best way to get the most computing stations per classroom. Another way to get more computing stations is to obtain a low cost local network, a device which will permit sharing of expensive computer peripherals. If you would have to obtain multiple printers, or disk drives, you could invest more in obtaining additional computer stations. Several sources of such networking devices now exist and the potential for more is high. Programming, word processing, and large group utilization of computers are three of the applications which will most definitely benefit from networking.

In conclusion, be sure to ask yourself two questions when setting out to acquire computer hardware. First, what are your computer literacy goals? And second, can low cost computing methods and equipment meet those goals? If you find a match, move forward, because you will have be joining one of few promising movements in education today: the computer revolution.

COMPREHENSIVE DATA MANAGEMENT SYSTEM (CDMS) for INDIVIDUAL EDUCATION PLANS and SPECIAL EDUCATION PROGRAM MANAGEMENT

Robert O. Guth, Ed.D., Director -
Pupil & Special Education Services
Fairfield Pupil Schools
Fairfield, Connecticut

INTRODUCTION

The CDMS represents a successful attempt to enter, store and produce a complete IEP with all of its components on a computer. In addition, the IEP becomes a data base to generate special education management reports. Some of the advantages of the system are as follows:

Time Saver: It is estimated that each Planning & Placement Team meeting can be reduced by ten minutes because the IEP can be written at the meeting using short numerical codes from a catalong of descriptors, goals and objectives, and criteria for success. A completed IEP includes all components to ensure compliance. Thus, staff members no longer have to spend more time with students and less time in meetings. Parents can approve the abbreviated forms at the meeting and then receive a completed IEP totally printed by the computer. All the information will be stored as a data base for management report generation and future use. Once the IEP is stored, the staff only have to make the changes and do not have to rewrite the document at the next review meeting.

Flexibility Continues: Despite computerized catalogs and standardized forms, the User will be able to add their own statements, goals and objectives, plus they will have the ability to write and store individualized comments and special notes. These statements can be written in a free form style without the need of a catalog. Since some curricula have similar objectives for each grade level, the system will allow the entry of levels. The User can edit the objective statement and will have the option to use criteria ranging from the very general to the most specific. The User can add, delete or change any file with a simple procedure.

Usefulness: Instead of the IEP being a form to meet mandates, it can now become a working, dynamic document quickly accessible, either on a visual screen or by hard copy. Changes are made with the word processing concept and it is not necessary to rewrite the entire IEP to produce a changed or updated copy. To make the IEP more useful, the system includes an optional statement of strategies, procedures or techniques if the User desires. This is not required for compliance but it gives the PPT the opportunity to review and consider all the options available in the school system.

Compliance Guaranteed: Because the IEP will use a standardized form and the input procedure will not allow the clerk to ignore or pass certain key pieces of information without a valid entry, the IEP produced by the computer will be in full compliance and will be ready for any Due Process challenge. It causes staff to adhere to your school system's procedures. The printing can be done in any style. For speed, a GE Matrix Printer can be used (200 characters per second) or for a letter quality document, a selection of type can be chosen through the use of a daisy wheel printer. The latter will produce an IEP that has the appearance of an individually typed document. Parents are impressed by the efficiency of the system and the professional appearance of the IEP.

EVALUATION

IEP Progress Report: The goals and objectives can be drawn from the IEP and produced as a quarterly progress report for use by the teacher. A User can have the computer store the results or have the report produced to be completed by the teacher with a handwritten copy attached to the IEP in the cumulative record. One caution that must be observed is the issue of when it is more efficient to complete a task by hand or when it is more prudent to maintain a computer system to accomplish the task. The task of completing quarterly progress reports may be more efficient to do by hand.

Test Scores: The computer will store the previous test score and can calculate changes. If a User desires, it can report them as averages for classes, programs or as a statistic on a future IEP. Again, this is an issue each User will have to decide. Calculating and reporting the change in test scores can cause the numbers to have an unreal priority.

MANAGEMENT TOOLS

Management tools can be generated from the data base created by the IEP. Some reports currently available are:

1. Class Roster: (general use) A report of the student's name, address, phone, parent name, listed by school, class or teacher, depending upon the User's desire.

2. Class Roster: (professional use) A report including the above information, plus test scores and other selected information for professional use. As long as the IEP is current, this and the former report will always be up-to-date.

3. Program Report: A list of all the students in each program, such as all the students whose IEP listed Physical Therapy, Speech, etc., as a program. The information will be listed by building, number of hours scheduled, and staff members assigned. This report is useful to determine the number and kind of staff needed and to check the efficiency of staff assignments.

4. IEP Expiration Date Report: Since IEPs can be written up to one year, a school system can stagger this expiration date so that all IEPs do not become due in the Spring of the year. On a monthly basis the report can give a list of IEPs due to expire in the next month.

5. Diagnostic Evaluation Expiration Date Report: Some state laws require that each diagnostic or psychological evaluation must be updated every three years. This report will give you, on a regular basis, all the diagnostic evaluations that will be three years or older as of the designated date.

6. State Reports: The system can generate the reports required by any state. The information on the IEP can be accessed and arranged in a special report or designed to be produced on pre-printed forms that are required by state mandate.

7. The Report Generator: The data base created by the storage of the IEP is available for the User to generate their own report. With minimal training and without the need for a professional programmer, a User can totally construct their own report of desired data to meet a particular need for a unique request.

8. Other Possibilities: The system will have the capacity to run other programs. For example, there is a Tuition Accounting Program. Each Special Education Program or business office must monitor the tuition paid to other agencies. This program will report tuition paid, tuition needed to be paid, and the total obligation. The program will contain a list and description of each facility educating your school system students and a description of each contract written. It will also report the statistics needed for the state report regarding Special Education tuition.

Another program available is a student projection system for use in changing enrollment that will assign students to various schools using size of class, number of classes, and school size as criteria.

EQUIPMENT

The present system for 1000 students can operate on a WANG 2200 VP using 56K of memory on a 10 mg disk drive. However, in the Fairfield Public Schools the Central Office found many uses for Word Processing and Records Management System and have upgraded the original equipment to 128K (capacity 256K) and added a multiplexer and three additional terminals (a maximum of 8 are possible).

GENERAL COMMENTS

Each report has a format that a User can alter or edit, using a very simple procedure to fit a particular need. In addition, reports can be generated for the entire data base or for a specific class or teacher. The system can be designed to meet each individual school system's particular needs. Each report can be generated at any time, either on a screen for a quick preview or a hard copy can be printed.

Confidentiality is maintained because the system requires the use of a password and knowledge of the entry procedure. Thus, unofficial personnel cannot use the system--far more secure than file cabinets.

The system can be a centralized service for the entire school system or each school within a system could have a printer and pterminal that would access the central CPU and disk drive through a telephone.

WANG Laboratories also have available a Word Processing Program and a Record Management System that would be compatible to the CDMS. For example, a letter to be sent to parents of all Learning Disabilities Resource students could be entered in the Word Processing Program and it would individually type and address each letter. If requested, it could also address envelopes. Since all of the above systems are compatible, the computer system can complete a wide variety of tasks and services not even contemplated.

CONCLUSION

The CDMS is functioning and, surprisingly, it has been fully accepted by staff and parents. Staff readily see the time-saving aspect and the elimination of repetitive, routine writing. Parents are impressed with a professionally prepared IEP as opposed to previous handwritten, and sometimes illegible, documents. In addition, the entire Central Office has benefitted from the system because each department has developed their own uses. The availability of a mini computer in the office has been an asset to everyone.

REFERENCES:

Equipment:
WANG Laboratories
One Industrial Avenue
Lowell, Mass. 01851
Phone: (617) 459-5000

Programmer:
New England Management Services, Inc.
240 Bradley Street
New Haven, Conn. 06510
Phone: (203) 787-3281

TAXONOMY OF INSTRUCTIONAL STRATEGIES
for
COMPUTER BASED EDUCATION*

Keith A. Hall
Ronald C. Comer
John A. Merrill

THE OHIO STATE UNIVERSITY

> . . . most students can attain a high level of learning capability if instruction is approached sensitively and systematically, if students are helped when and where they have learning difficulties, if they are given sufficient time to achieve mastery, and if there is clear criterion of what constitutes mastery.
> (Bloom 1976, p. 4)

Perhaps some of the most powerful suggestions for improving the quality of CBE courseware can be derived from observing the origins and characteristics of existing courseware. Historically, when involvement in CBE was restricted to a few (maybe 12) centers across the country, considerable improvement in the quality of courseware could be observed over time as the developers became more experienced and as knowledge of the characteristics of effective courseware began to accumulate and impact new courseware development. However, courseware development is now the purview of anyone with a microcomputer - many of whom are ill prepared to develop quality materials.

Teachers and "computerists" are frequently the authors of CBE courseware. Teachers primarily focus on the act of teaching and the role of the teacher in the process. Typically their experience includes "explaining and telling" 20 students simultaneously. When discussing questions of instructional strategy and design, their most common utterance is "I could . . . " Their resulting courseware can often be characterized as an on-line lecture which the student observes.

Computerists whose focus is on "computer power" are generally experienced programmers with educational experience as students and perhaps as the parent of students. In discussing CBE courseware they can be identified by their more common utterance "The computer could . . ." The computerist's courseware is characterized by sophisticated pyrotechnics which the student observes.

The teacher produces courseware which flows from the teacher, through the computer (with little alteration) to the learner. Input, processing, and output are concepts of little consequence. In contrast, the computerist's courseware places heavy emphasis on the processing done by the computer and acknowledges that the computer provides output to the learner and receives input from the learner - generally in the form of a single key press to signal the computer to proceed.

Although the authors' perspectives are subtle and sometimes un-noticed, the impact on the quality of courseware is dramatic. Research has consistently shown that deep cognitive processing by the learner is prerequisite for learning. Deep cognitive processing does not necessarily take place (although it might for some individuals) while the learner 'watches' a program execute on a computer. Rather, asking questions which cause the learner to manipulate content material in ways which he has not been directly taught is a more reliable method of causing deep cognitive processing. Questions which cause the learner to perform at successively higher and higher levels of cognitive activity are the hallmark of quality CBE materials.

At the outset it was hypothesized that effective instructional strategies should consider the structure of the content to be learned, the level of learning desired, and individual differences among learners. Our research, however, has cast doubt on the use of individual differences as a key element in instructional strategies; but rather the kinds of questions presented to learners are an extremely critical variable to be included.

STRUCTURE OF KNOWLEDGE

Almost all words that convey meaning are used as concepts. Whether we are teaching the most basic facts associated with a given content area or the most complex theoretical constructs we are usually teaching concepts. Of particular importance to instruction is the role ascribed to concepts by Klausmeier and Associates (1979):

> Concepts as mental constructs are the critical components of a maturing individual's continuously changing, enlarging cognitive structure and are the basic tools of thought.
> (p. 7)

Two alternative approaches may prove useful for instructional development: (1) learning concepts by attribute isolation and identification (Bourne, 1974; Haygood & Bourne, 1965; Bruner, Goodnow, & Austin, 1956) and (2) learning concepts by prototype (clear case or best example) (Garner, 1970; Reed, 1972; Shepp, 1978, Wilson, 1963). The use of one approach or the other, or perhaps a combination of the two approaches, may be greatly influenced by the degree of specificity of concepts in the discipline. Some disciplines may lend themselves very readily to attribute isolation (e.g., mathematics and science) and other disciplines (e.g., social studies and the humanities) may be more amenable to describing examples.

Concept Learning By Prototype

Wilson (1963) proposed that the following elements should be presented sequentially until the learner has grasped the concept: the central meaning, a model case, contrary case, related case, borderline case, invented case, social context, underlying anxiety, practical results, and results in the language.

The prototype (analytic philosophy) approach may be particularly useful for social studies and the humanities because of the abstract concepts and the accompanying lack of specific definition found in those areas. This approach might also be useful as learners mature and advance their understanding to higher levels of abstraction, e.g., as the meaning of "freedom" is refined and expanded from the age of 6 to the age of 15.

Concept Learning By Attribute Isolation

Teaching concepts by attribute isolation is supported by data from a number of researchers (Clark, 1971; Entwistle, 1978; Gagne, 1962; Klausmeier, Ghatala, & Frayer, 1974; Tennyson, Tennyson, & Rothen, 1980; Tennyson, 1980; Tennyson, 1981; Tennyson & Park, 1980; and Tennyson & Rothen, 1979). The procedure usually includes presentation of the concept definition (attributes), an expository presentation of instances of the concept with help in isolating attributes, an inquisitory presentation of instances followed by attribute isolation feedback.

Implications for Courseware Development

Concept learning by prototype and by attribute isolation have been documented experimentally as effective techniques, although much of the data was gathered in classroom and experimental environments which did not incorporate individually adaptive instructional strategies. On the other hand, much existing, adaptive CBE courseware has not incorporated the empirically validated strategies of concept learning. Adapting the experimental findings of concept learning to the adaptive environment of CBE should provide a quantum leap in the quality of CBE courseware.

LEVELS OF LEARNING

Each time a decison is made to provide instruction to a learner, an accompanying decision of level-of-learning should be made. Several taxonomies of instructioal objectives have been developed, but Bloom's Taxonomy of Educational Objectives was selected because it exhibited the following characteristics: (1) validated comunicability, usefulness, and suggestiveness, (2) emphasis on types of learning tasks generally encountered in instructional environments, (3) broadly accepted by educators, (4) quality documentation, and (5) congruence with the structure of knowledge and the tasks learners are expected to perform within the structure of knowledge.

Implications for Courseware Development

Bloom's taxonomy will provide three important functions for courseware improvement: (1) establish a standard range of levels-of-learning (objectives), (2) raise authors' consciousness of the range of instructional objectives and the improved quality of courseware when higher levels are included, and (3) provide course authoring prescriptions to produce courseware which will cause learners to perform at each level.

LEARNER DIFFERENCES

> . . . Historical emphasis on individual differences has obscured efforts to deal with the educational problems that are more concerned with the quality of interactions between individuals and their educational/social environments.
>
> (Bloom, 1976)

Theoretically, students with high or low verbal, mathematical, spatial reasoning, or mechanical comprehension abilities should react differentially to alternative instructional treatments in the context of specific content. However, Cronbach and Snow (1977) found that special abilities generally do not interact with treatment, whereas general ability is much more likely to do so. Snow (1977) noted that ATI results to date do not make instructional theory impossible, but may make general instructional theory impossible, i.e., instructional treatments applied in a specific setting and at a specific time with a specific student or group of students, may be effective but the same instructional treatments may not generalize to all settings, students, instructors, and times (Snow 1977).

Most educators would agree that learning is a highly complex process that is affected by a number of interacting variables. Individual variables with sufficient power to enable us to design instructional materials around them have not been identified. We do know though that learning is affected by a number of variables including prior knowledge, age, intelligence, social experience and competence, creativity, and motivation. Traditionally, adjustments for these variables have been made at the curriculum development stage to insure that learners and instructional expectations are within a suitable range. Once instruction is started, little attention is paid to systematically adapting to learner differences.

Research to date has not enabled us to knowingly devise instructional strategies to match unique characteristics of individual learners. The research leads to the following postulates for designing instructonal strategies:

- Developmental stages are more related to curriculum development than to instructional development and do not require close attention if authors of CBE courseware are experienced in working with the targeted audience.
- The ability of learners to adapt to instructional environments is more powerful than differential instructional strategies.
- An adaptive instructional strategy which makes 'real-time' adjustments to the demonstrated performance of each learner has in effect dealt appropriately with the individual differences of learners without the need to categorize or label those differences.

QUESTIONING TECHNIQUES

> . . . The question is perhaps the primary tool by which the individual processes information regardless of the diversity of his procedure.
>
> (Hunkins 1976, p.4)

The role of questioning in instruction has been a significant concern among educators for well over a half-century. Piaget and Dewey (Newton, 1978) proposed intellectual development through questioning, and over the years educators have advocated the use of effective questions to stimulate thinking among learners (Burton, 1929; Taba, 1967; and Hunkins, 1968). Despite the continuing interest, Gall (1970) concluded that in the last fifty years there has been no essential change in the types of questions teachers emphasize in the classroom. About 60% of teachers' questions require students to recall facts, 20% require students to think, and the remaining 20% are procedural. So much focus has been on the characteristics of questioning practices that little is known about student behavior caused by different types of questions (Buggey, 1972; Dunkin & Biddle, 1974; Hunkins, 1972; Rosenshine, 1971; Ryan, 1973; Savage, 1972; and Tyler, 1972).

Even though only 20% of the questions asked in the classroom require the learner to think (Gall, 1970), many persons believe that a major purpose of education is to help learners develop creative and critical thinking abilities. Watson and Glaser (1964) defined critical thinking as the composite of attitudes of inquiry; knowledge of inferences, abstractions, and generalizations; and skills in applying the above. Dressel and Mayhew (1954) identified five abilities related to critical thinking, ranging from the ability to (1) define a problem (analysis), (2) draw valid conclusions (synthesis), and (3) judge the validity of inferences (evaluation). Questioning learners (and encouraging learners to question also) appears to be the accepted process for achieving the higher cognitive goals of education. If instructional objectives require learners to perform throughout the range of Bloom's taxonomy, then clearly the instructional environment must provide opportunities (questions) which cause learners to practice appropriate performances.

Ryan (1973) dichotomized questions into low levels (recall category) and high levels, the latter including six categories: process, relationship, application, educated guess, synthesis, and opinion. Ryan found that high level questions were more efficient than low level questions for moving students toward both low and high level understandings because high level questions require learners to recall facts and data, then analyze and synthesize the data before responding to higher level questions. Ryan's work suggests that the level of questions defines the level of learning resulting from the instruction, and therefore questioning should be conducted purposefully and systematically. Andre (1977) provided a link between Ryan and Bloom by defining the behaviors required of the learner at each level of Bloom's taxonomy:

Knowledge--repetition of information in the form it was presented. Questions at this level are often labeled factual questions.

Comprehension--recognition or production of some paraphrase of material presented in instruction.

Application--use of presented information in some new situation. Application could include recognizing new examples of a concept or using a principle in a problem solving situation.

Analysis--decomposing a given situation into its component parts and analyzing their relationships. Analysis typically requires the use of some previously taught scheme to decompose the whole, e.g., the student may be told to analyze a short story.

Synthesis--requires production of some product given appropriate elements, e.g., writing a short story.

Evaluation--requires judgments about the value of information, concepts, or ideas relative to some goal or purpose.

Andre proposed that a taxonomy of questions should include 1. objective classification of questions, 2. objective mechanism by which questions of each type could be constructed, and 3. psychological and instructional validity for each category of questions.

Implications for Courseware Development

If knowledge is produced in response to questions, and new knowledge results from the asking of new questions (Postman and Weingartner, 1969), the ability to form concepts and the ability to formulate and respond to questions are related.

CONCLUSIONS

The review of literature has modified and reinforced some of our earlier conjectures on a taxonomy of instructional strategies. The usefulness of a collection of mutually exclusive strategies for describing courseware now seems to be a moot point. Certainly there are broad identifiable strategies such as computer managed instruction, interactive instruction (which generally includes drill and practice and tutorial instruction), and instructional simulations. But these strategies do not describe characteristics of the courseware which are central to the questions of instructional quality we should be asking about courseware -- (1) level of learning provided (rote memory, concept attainment, or concept production), (2) structure of the content, or (3) the level of cognitive processing required of the learner.

A preponderance of literature suggests that most learning is conceptual, and that differing levels of conceptual learning can be defined across the hierarchy of instructional objectives defined by Bloom's taxonomy. Two approaches to teaching concepts have emerged from the literature: (1) prototype presentation and clarification which may lend itself well to the arts, humanities, and social studies where concepts and their attributes are not always clearly defined and (2) attribute isolation and discrimination which may lend itself well to highly structured fields such as mathematics and science. Providing both approaches will accomodate authors from a variety of backgrounds.

Regardless of the approach selected by a courseware author, probably the single most important factor in producing courseware of a high educational quality is the nature of questions which are presented to the learner. The taxonomy must incorporate a taxonomy of question types which cause learners to respond throughout the range of Bloom's taxonomy.

---

*Research sponsored by Control Data Corporation.

EVALUATING AND SELECTING A BUSINESS BASIC

James M. Hoey
Potomac State College
of
West Virginia University

INTRODUCTION

BASIC was developed at Dartmouth in 1964 by John Kemeney for the purpose of providing students with an easy to learn low cost programming language. BASIC has evolved to where every hardware and software vendor has added unique enhancements to BASIC to the point where the only thing standard about BASIC is the lack of uniformity among BASIC languages. Many computer manufacturers and some private software firms have also added a super set of enhancements to BASIC to be used in a business environment. These versions are often referred to as BUSINESS of COMMERCIAL BASIC.

A close examination of available BUSINESS BASICs reveals a great disparity in function and capability among the various versions of BUSINESS BASIC.

The rhetoric used to describe BUSINESS BASIC features terminology such as comprehensive, flexible, advanced and powerful. This terminology leaves tremendous latitude for actual interpretation by someone considering acquisition of a BUSINESS BASIC language processor. The following discussion examines many of the functional areas of the BASIC language and provides some insight into some desirable features that a good BUSINESS BASIC should have in order to efficiently develop high quality, easy to use Business software.

For purposes of this paper, business application shall be defined as those type systems and applications which are an integral part of daily business activities. These activities include: Payroll, Accounts Payable, Inventory, Accounts Receivable and many others. These activities are characterized by large amounts of data being contained in data files for subsequent updating, processing, reporting, and archival storage.

The format of this paper shall list various desired specifications and briefly offer some justification for these specifications.

A. GENERAL CHARACTERISTICS

ABILITY TO USE EXPANDED VARIABLES OF 6 OR MORE CHARACTERS ALL SIGNIFICANT.

This feature enables programs to be easy to read, and subsequently easier to change.

ABILITY TO USE UP TO 255 (OR MORE) CHARACTERS ON A SINGLE LINE OR VIA CONTINUATION.

This is necessary in order to process longer statements caused by the longer variable names and to process long I/O lists encountered with a read (file), thus eliminating a REREAD (file) statement.

ABILITY TO APPEND A LABEL TO BE USED IN IDENTIFYING A STATEMENT, SUCH AS,100 HEADER.

This makes branching statements easy to use and improves readability of program.

ABILITY TO MERGE PROGRAMS

This enables reusable program

modules to be easily integrated into a new program.

ABILITY TO PROCESS UPPER/LOWER CASE CHARACTERS.

ABILITY TO PROCESS INTEGER AND FLOATING POINT NUMERIC VARIABLES WITHOUT SPECIAL NAMING CONVENTIONS.

Integer and floating point variable should not require special variable naming conventions.

ALL VARIABLES (NUMERIC AND STRING) SHOULD BE AUTOMATICALLY INITIALIZED TO ZERO AND BLANKS AT INITIAL RUN TIME.

This eliminates the necessity of program code to perform this function.

ABILITY TO DEFINE SPECIAL FUNCTION KEYS WHICH ARE ACCESSIBLE TO PROGRAM CONTROL.

ABILITY TO HANDLE COMPLEX ARITHMETIC ASSIGNMENT STATEMENTS.

This eliminates the necessity of constructing several statements to accomplish what could be done in one statement.

B. PRINT OUTPUT

ABILITY TO EASILY FORMAT PRINTED OUTPUT LINES VIA A USING TYPE STATEMENT.

This permits easy development of specially designed reports.

ABILITY TO EDIT PRINTED OUTPUT.

This includes special format descriptor characters that should enable zero suppression and numeric editing (sign, comma, CR/DB, decimal points).

ABILITY TO CONTROL LINE PAGE EJECTS.

This enables the user to easily skip lines and locate top-of-page without extensive programming. Print statement should be able to detect page overflow condition.

ABILITY TO CONTROL VERTICAL AND HORIZONTAL PRINTER SPACING UNDER PROGRAM CONTROL.

User should be able to easily change the vertical and horizontal spacing characteristics of printed output.

ABILITY TO PRINT ARRAYS OF DATA.

C. FILE HANDLING

Business functions have a heavy dependence upon the manipulation, (adding, deleting, changing and sorting) of information contained in data files. Therefore, good file handling capability is an extremely important facet of a BUSINESS BASIC.

DATA FILE ACCESS METHODS SHOULD INCLUDE SEQUENTIAL, RANDOM AND INDEXED SEQUENTIAL (KEYED).

Good file handling greatly facilitates programming effort and system operation particularly indexed sequential files (ISAM). ISAM files permit random access based on any desired key, ie; Social Security number, name or any user developed key. This is extremely important as business type data must often be accessed by a non-numeric type key.

ISAM FILES MAXIMUM KEY LENGTH SHOULD BE AT LEAST 12 CHARACTERS.

ISAM SHOULD (OPTIONALLY) PERMIT DUPLICATE KEYS AND ALSO PERMIT MULTIPLE KEYS PER FILE.

Duplicate key and multiple keys further enhance the use of ISAM files as a single file may now be accessed by different key fields.

ISAM KEYS SHOULD AUTOMATICALLY BE REORGANIZED AT OPEN OR CLOSE TIME.

In order to function effectively ISAM files should be reorganized periodically. Preferably this should be done automatically and should be transparent to the user.

DATA FILE RECORDS SHOULD BE ABLE TO CONTAIN MIXED DATA TYPES, INTEGER, REAL AND STRING.

THE USER SHOULD HAVE THE CAPABILITY OF USING FIXED FIELD RECORD FORMATING BASED ON A FORMAT DESCRIPTORS-SUCH AS WRITE USING.

DATA FILES SHOULD EXTEND DYNAMICALLY WHEN FULL.

NULL FILES SHOULD BE PERMISSABLE.

This eliminates having to write dummy records into an empty file.

FILES SHOULD BE ABLE TO HAVE RESTRICTED ACCESS FILE PROTECTION.

PROGRAM CONTROL SHOULD BE ABLE TO ACCESS VARIOUS CURRENT FILE ATTRIBUTES, SUCH AS RECORD NUMBER, KEY POSITION, KEY LENGTH, ETC.

PROGRAM CONTROL SHOULD HAVE THE ABILITY TO AUTOMATICALLY IDENTIFY END OF FILE.

End of file should not generate an error condition as a method of detecting end of file.

D. ERROR HANDLING

First time end users are highly susceptible to making errors, therefore, in order to prevent abnormal program termination, user errors should be able to easily be trapped under program control.

ABILITY TO TRAP DATA CONVERSION ERRORS DURING ARITHMETIC OPERATIONS.

ABILITY TO TRAP DATA CONVERSION ERRORS DURING I/O OPERATIONS AND PROVIDE IDENTIFICATION OF THE BAD DATA ELEMENT.

ABILITY TO ACCESS ERROR CODES UNDER PROGRAM CONTROL.

ABILITY TO PROVIDE EXITS FOR VARIOUS TYPE OF ERRORS.

E. BRANCHING INSTRUCTION

ABILITY TO USE STRUCTURED STATEMENTS SUCH AS IF...THEN...ELSE.

Structured IF...THEN...ELSE statements improve readability and enhance logical construction by not requiring unnecessary program instructions in order to accomodate the desired logical relationships.

ABILITY TO USE COMPOUND LOGICAL OPERATORS SUCH AS AND/OR.

This enables the user to test for several logical conditions in one statement.

ABILITY TO EXECUTE STATEMENTS OTHER THAN GOTO AS THE TRUE CONDITIONAL BRANCH SUCH AS IF A=B THEN PRINT "AMOUNT IS", TOTAL.

This eliminates unnecessary branching and improves the program's logical flow.

F. FULL SCREEN HANDLING

Full screen processing is the ability to paint the screen with a form and permit the user to easily enter data into the form via cursor movement.

ABILITY TO EASILY CONSTRUCT SCREEN MASKS AND EASILY MANIPULATE THE CURSOR FOR DATA INPUT OR UPDATING VIA BASIC TYPE INSTRUCTIONS.

ABILITY TO ASSIGN SPECIAL DISPLAY ATTRIBUTES SUCH AS BLINKING, HIGHLIGHT, UNDERLINE, REVERSE VIDEO TO DESIRED SCREEN LOCATIONS.

G. SYSTEM FUNCTIONS (INTRINSIC)

ABILITY TO TRIM OR PAD SPACES IN STRINGS.

ABILITY TO SORT ARRAYS (ASCENDING, DESCENDING).

ABILITY TO SEARCH ARRAYS FOR A VALUE.

ABILITY TO CONVERT BETWEEN STRING AND NUMERICS.

ABILITY TO ROUND VALUES.

ABILITY TO SEARCH A STRING FOR A DESIRED SUBSTRING.

ABILITY TO SUBSTRING A STRING.

ABILITY TO ZERO ARRAYS (MATRIX) OR SET AN ARRAY TO A CONSTANT VALUE.

ABILITY TO REDIMENSION ARRAYS.

ABILITY TO REPLACE OR INSERT CHARACTERS IN A STRING.

H. PROCEDURE FILES

Procedure files permit various BASIC control commands to be executed automatically from a special type with little or no operator intervention required.

ABILITY TO ESTABLISH FILES TO AUTOMATICALLY CONTROL SYSTEM PROCESSES.

ABILITY TO NEST SUB-PROCEDURES THAT MAY BE CALLED FROM A MAIN PROCEDURE.

ABILITY TO MAKE LOGICAL BRANCHES WITHIN A PROCEDURE BASED UPON AN EXTERNALLY ASSIGNED VALUE.

I. UTILITY PROCESSES

Various utilities should be available to merge, copy, sort and reorganize data files. These utilities should be accessible from a procedure or from program control.

SUMMARY

Increased pressures by end users of small computer systems for systems that are easy to use demands that the BASIC language processor contain the capability that will enable the user to efficiently develop these type of systems. The selection of a powerful business BASIC is a key prerequisite in achieving this objective. To be a truly powerful business BASIC it must contain many of the aforementioned capabilities. More importantly, it must contain a powerful input/output capability; whether it be I/O to the screen, I/O to the printer or I/O to data files. Without powerful I/O capability it is virtually impossible to develop quality business software using the BASIC language.

331

OBSERVATIONS ON THE DYNAMICS OF K-12 TEACHER INVOLVEMENT WITH MICROCOMPUTERS

Dorothy H. Judd, Ed.D.
College of Education
Northern Illinois University

and

Robert C. Judd, Ph.D
College of Business Adm.
Governors State University

The literature of social change regarding diffusion of innovations and their adoption by specific target user groups has been extensive. Relatively less has been written about the diffusion of the innovation represented by the instructional use of microcomputers in the K-12 classroom. This paper considers the factors involved in rate of diffusion and adoption of the microcomputer for instructional use.

An informal assessment of behaviors and attitudes of teachers have been collated from (1) institute days or teacher work-shops, and (2) graduate credit courses in classroom use of microcomputers offered by two different universities. The setting of the informal research reported in this paper begins with two summer session courses of 1980 offered in the western suburban elementary school district #181 of Hinsdale. These two courses, offered by Governors State University, carried two semester hours of graduate credit. The fall and winter of 1980-81 provided opportunity to observe workshops at Harlem Unit School District 122 at Rockford, a seminar on microcomputer use in teaching special education students sponsored by the National Science Foundation and Chicago State University, an institute day session in Hinsdale #181, and a seminar session for principals on microcomputer use sponsored by Northwest Educational Consortium. Two more observational inputs were afforded by a three semester hour graduate credit course offered by Northern Illinois University and held at Deerfield High School, a northern suburb of Chicago, in the fall and spring semesters of 1981-82. An institute session on microcomputer use in elementary schools of District 132, Calumet Park, a southern suburban location, concluded the observations. Two of the graduate credit courses provided a hands-on opportunity with two brands of microcomputers. The other two graduate credit courses offered a hands-on experience with only one brand of microcomputer. All of the teacher institute day or seminar sessions involved a hands-on experience with one brand of microcomputer.

Examples of the behaviors and attitudes recorded in the process of the present informal research include:

- *Responses by teachers as they experienced using microcomputers in an instructional mode.
- *Skill development strategies that appeared to be effective with teachers.
- *Attitudes of K-12 teachers toward their involvement with microcomputers for instructional use.

The literature on the diffusion and adoption of innovations stresses that those most likely to adopt anything new will be those with more education and they will tend to adopt the "new" earlier than their peers.[1] Despite this general finding, there is considerable hesitancy among the ranks of teachers to adopt the use of microcomputers. Building on the work of Rogers and Shoemaker, Martellaro has proposed that there are five aspects of an innovation that impact its likelihood of adoption.[2] The five attributes are (1) relative advantage, (2) complexity, (3) trialability, (4) observability, and (5) compatability. The experience of the present authors suggests that factors 1, 3, and 4 can be addressed successfully by inservice programs and extension courses.

The issue of complexity and compatability, like the issue of beauty, is found in the eyes of the beholder. The teachers who had trouble with keyboard input were not just fumble fingered, they felt there was a complexity to the concept of touching a key, seeing the letter appear on the monitor, and knowing that because of their act, the computer would follow their command. While there may be an element of finger dexterity involved in teacher difficulty with keyboard input, a strategy that seemed to help was to have the teacher make keyboard inputs in a word processing mode. The use of a keyboard to create the same kind of sentence form the teacher was used to creating for a quiz for his or her class on a typewriter seemed to be easier to accomplish than to key in commands in BASIC, or even in the PILOT language. The responses of teachers and the behaviors that represented their response to the microcomputer took two forms. On the one hand, a teacher would have no problems in keyboard input, in inserting diskettes into the disk drive, or in running the programs. On the other hand, a teacher would fumble with keyboard input, be unable to get the diskette into the disk drive correctly, and no matter how hard they tried the program would not run. typical of this latter situation was a teacher who professed great enthusiasm for the course and for microcomputer use in classroom instruction. Yet, this teacher, after great difficulty with keyboard input of a short program in BASIC, finally got it to run, but then was almost in tears because she could not get the program to save on her diskette. The difficulty proved to be that she had removed her diskette from the disk drive and had forgotten that it was not in place when she was ready to save her laboriously typed program code. There were other episodes, not as dramatic or foolish, but evidencing almost a desire to fail at what the teacher was about.

The most significant finding that the current informal research has is that those who want to succeed, do; those who do not want to succeed, don't. The teachers involved may not even recognize, let alone admit that they do not want success. Their behavior in fumbling, in failing to follow the precise routine required by the microcomputer in turning it on, in the keyboard entry of a program, and in careful attention to details of diskette handling make the outcome of their experiences with a microcomputer less satisfying than they profess to seek. The observations the authors made tend to demonstrate that response to the microcomputer is not a function of age. Some of the most successful teachers in these classes and institutes as well as the few that had difficulty were among the oldest present.

Solntseff of McMaster University in following up on the work of Martellaro outlined a specific model of what a teacher does in being a teacher and how the use of a microcomputer addresses those needs. Based on the experience of the authors in the observations reported in this paper, there is much in the Solntseff model that can be used to advantage by educators who are interested in promoting more painless and more effective interface between teachers and the microcomputer as an instructional tool. The model Solntseff proposed involves: (1) scheduling of resources, (2) organization of information, (3) document preparation, (4) storage of information, and (5) transmission of information.[3] The strategy of having teachers input the words of a quiz on the microcomputer, commented upon earlier, meets the requirements of (3) above. Preparing tests, worksheets, and other paper documents is part of teaching. Teachers can relate to that, and thus to the microcomputer. Further, there was special enthusiasm among the teachers observed in this research as to the ability of the microcomputer to provide an item analysis of test items missed by students, and the ability of the microcomputer to provide an evaluation of the reading difficulty of a book. In the latter instance, even those who often had trouble with keyboard entry tasks were able to enter the three 100 word passages on the keyboard. They were involved in dealing with words, and their keyboard task was not that different from using a typewriter. They were pleased that the microcomputer could count letters, syllables, words, and sentences and give them an evaluation of the estimated reading difficulty of the book.[4] These reading evaluation microcomputer techniques also satisfied the Solntseff model element #2 on organization of information.

Shostak reported that a high school English department chairperson in Florida became the moving force behind the use of microcomputers in her school, because she had the need for solving a complicated schedule in her new department program for teaching English.[5] When she was told it could not be done, she set about learning enough about computers and programming so that with the aid of a student programmer, the schedule she sought was completed. She went on to learn about microcomputers and computer assisted instruction (CAI) and today has a most successful remedial lab for her English students utilizing ten microcomputers.

The authors did not encounter other proofs of the Solntseff model in the observations involved in the current study. However, each of the five aspects of teaching that Solntseff lists contributes to the acquisition of more positive teacher attitudes toward microcomputer use. Although the actual attitude of some teachers about an innovative tool may be different from their professed stance, most of the teachers were decidedly positive in their response to the microcomputer as an instructional tool. The conclusion of the authors is that as teachers become familiar with the microcomputer, their enthusiasm increases. Finding new ways of planning and implementing more effective instruction in the curriculum for tomorrow becomes a reality through use of a microcomputer.

ENDNOTES

[1] Everett M. Rogers and F. Floyd Shoemaker, Communication of Innovations, 2nd ed., (New York: The Free Press, 1971), pp. 107-8, 186, 241.

[2] Helena C. Martellaro, "What's Keeping Computers Out of the Classroom?" The Education Digest, January, 1981, pp. 27-29.

[3] Nick Solntseff, "What Do We Tell the Administrators?" Creative Computing, March, 1981, pp. 100-2.

[4] Dorothy H. Judd, "Avoid Readability Formula Drudgery: Use Your School's Microcomputer," The Reading Teacher, 35:7-8, October, 1981.

[5] Robert Shostak, "Computers and Teaching English," The Computing Teacher, January, 1982, pp. 34-35.

MANAGING INFORMATION FOR NEW KINDS OF LEARNERS

Greg Markovich and Joan Malling

America's colleges face serious problems in the decade ahead, indeed, many of the problems are already realities. The decline in the 18 to 22 year-old age group, changes in funding sources and uncertainties in the economy threaten the very existence of many colleges. Creative solutions, however, are being developed, especially among the small liberal arts colleges.

Adult college, weekend college, evening division, non-traditional educational experiences--these are among the ways in which traditional colleges are redirecting their resources. These traditional colleges are now serving new kinds of learners, and the new "learning majority," consists of adults.

Pioneer among these repositioning institutions is Marylhurst College, which made the transition in 1974 from a traditional undergraduate program for young women to Marylhurst College for Lifelong Learning. Accredited to award the B.A., B.S., and B.Mus., Marylhurst provides educational services to an almost equal number of degree and non-degree students. Based on a unique approach to the delivery of services to suit the schedules of working men and women, and using experiential learning approaches, the entire institution is now devoted to the lifelong learner.

Marylhurst has been perceived from the beginning in 1974 as a trail-breaker, and much of the foundation support toward bringing this new institution into operation has come from widely respected national and regional foundations, such as the W. K. Kellogg Foundation and most recently the M. J. Murdock Charitable Trust. The Murdock Trust's grant of $30,000 in February, 1981, funded Phase I of a Marylhurst study to determine information needs for this new kind of educational organization and information to assist the college in responding to the new learners. This grant came about because the college realized that its historical (and rather traditional) approach to information collection, dissemination and utilization was no longer appropriate.

With new educational environments--scheduling, faculty staffing, part-time enrollments, documentation of prior learning in non-academic settings--information needs are new and different. The old "shoe-box" full of information just doesn't work any longer. The need for information and the dissemination process must be responsive to the new academic cycle, methods and most of all the new learners. Phase I of this project set out to discover these needs.

During this phase we realized two very important working assumptions that are now guiding the project, the first is:

"That all information systems activity must provide a benefit to the learner and that the learner must be at the "Center" of our activities. This "Centering" activity includes the adjustment of processes, forms and attitudes to better meet the needs of the learner."

The second working assumption that we realized has to do with the definition of MIS. What people usually think of when they first hear the term Management Information System is computers--basically putting data into a computer and getting certain kinds of printouts and displays of information out. What we learned is that this type of definition is too narrow and our new working assumption (definition) is:

"Management Information Systems are people, processes, forms and data working together to provide information to make decisions that further the mission and objectives of the institution and reinforce its commitments to learners."

With these working assumptions as our guide our list of activities and accomplishments of Phase I range all the way from removing counters (like in the Registrar's office), consolidating offices, eliminating extra forms, tearing down walls and designing manual and computer based systems. Needless to say the impact upon the institution has been noteworthy. The theme of learner-centeredness has become more vibrant and visible. Both teaching faculty and divisional chairpersons are looking for more information on learners, top administrators are seeking new statistical representations. All-in-all information management has a new meaning. We have realized that we can be systematized and even more personalized. Information uses has changed. Class rosters now contain telephone numbers. Information on previous educational experiences is being made available. Improvements in evening registration include a staff assistant designated to follow-up on incomplete registrations.

Essentially this Phase enabled us to conduct a needs analysis, review the delivery of information, and formulate a preliminary design for a new information system. This general systems design document will provide us with guidance for the next phases of the project. The total system has three subsystems namely: Learner Centered Information (includes historical Registrar data and new adult development data); Faculty Assistance Information (a total information system for division chairpersons and others, that includes advising worksheets, complete learner information and financial information) the third subsystem has two modules one dedicated to directions (i.e. institutional advancement and admissions including market analysis) and the other module concentrates on management (business office, planning and budgeting operating in an interactive mode). These subsystems will be able to provide data to answer questions like:

·What are the majors of students in this class?

·What is their previous educational background?

·How many are taking other courses from other colleges?

·What is the total tuition generated by Major by class?

·Do liberal arts students consume other courses?

·What is the break even point on a course?

·If a course is cancelled will the student withdraw?

·What is the previous learning experience of these students?

·What is the rate of return on display advertisng?

·How many inquiries does it take to produce 1 admission?

·Does advertising placement (day) impact rate of return?

·What is the state of the budget by division?

Based upon the general systems design that has been created it appears that the supporting hardware/software will be mini/micro, distributed, with a data base management system foundation. This configuration will enable the institution to strategically distribute its information resource. The needs analysis revealed that the information resource needs to be available to the faculty as easy as the administration.

Having completed this Phase we are now moving into Phase II, that is designed to operationalize our needs especially those that provide more information about (and for) the learner. To provide initial support to Phase II the institution has selected an IBM DataMaster/23 with their Data Base Management System (BRADS/3). This hardware has already enabled the college to begin building a data base that is user friendly. Reports are already being generated to support the information needs of the faculty as well as the administration. One report displays educational patterns and associated costs. We envision a data base of learner information which teaching faculty, division chairpersons and administrators will have access to. We see a set of reports depicting a "snapshot" of the learning environment and most of all we see improved education of our learners.

In summary, this project is designed to create an information system that is proactive to adult learners rather than reactive and that centers upon the learner and faculty information needs.

As previously mentioned the activities cited herein are the result of the generosity of the M. J. Murdock Charitable Trust for which the college is grateful. For further information contact Ms. Joan Malling at Marylhurst College.

Computer Literacy Education for K-6 Teachers

Charles C. Philipp, Department of Staff Development
Judie Muntner, Gifted and Talented Teacher
Patricia Cutlip, Computer Related Instruction
Montgomery County Public Schools
Rockville, Maryland 20850

To meet staff needs for computer literacy education, Montgomery County (Maryland) Public Schools developed an in-service course for K-6 teachers. The Concerns-Based Adoption Model from the Research and Development Center for Teacher Education, University of Texas at Austin was used as framework.

THE PLAN: Assumptions, Background, Comments

- Computer literacy includes skills, concepts, and attitudes.
- Computer literacy is a process, not an event.
- Classroom teachers are primary change agents in the schools.
- Computer literacy education for students must be linked with computer literacy education for teachers.
- Teacher education, equipment acquisition, instructional materials evaluation/acquisition, and curriculum development are the essential, interdependent components of a computer literacy program for students.

These five assumptions form the basis of a computer literacy program now underway in the Montgomery County, Maryland Public Schools (MCPS). This effort is different from previous program development efforts in that the idea of evolutionary change involving classroom teachers is playing a central role. This paper is about the K-6 teacher education component of this program--planning, developing, implementing it.

Montgomery County lies adjacent to Washington, D.C. The school system has 95,587 students with 5,282 classroom teachers assigned to 177 schools in a geographic area of 500 square miles. Today, like most school systems across the nation, MCPS hires relatively few new teachers each year. Even when recent graduates are hired, the teacher training institutions are not yet turning out computer literate educators. These two factors mean that curriculum improvements must be supported by local staff development programs which recognize that new programs for students will not be successful until individual teachers make a personal commitment to change.

In learning to teach students about computers and how to use computers, teachers must be given an opportunity to explore this new technology and to develop an understanding of its power and potential. As a practical matter, this means that teachers initially want to obtain some knowledge about how to program computers. Learning about programming is a reasonable first step for teachers because it provides a way for them to experience a sense of control and to gain an understanding of the kind of logic required for communicating with computers. Another reason for teachers to gain a basic understanding of programming is that it enhances their ability to evaluate courseware, judging it in the light of what a computer can and cannot be made to do in formatting text, evaluating responses, branching, pacing, recordkeeping, graphics.

This observation about programming leads to a second emphasis in teacher training: teachers must learn to make judgments about the use of technology. A viable computer literacy program for students will arise from a blending of computer technology with existing educational goals and practices. However, simply because something is possible with the computer does not mean that it ought to be done. Teachers must understand the ways in which the computer can be used as an instructional tool and must be able to select those uses appropriate to the academic disciplines and the individual students they teach.

The idea of evolutionary change has a central role in how Montgomery County teachers are introduced to computer technology. In the long run, it is not helpful to have teachers learn about only one particular computer literacy curriculum because over the next ten to fifteen years there will be many curricula. Applications in classrooms will become diverse. Programming languages and courseware packages of far higher quality than those now available will cause teachers to re-evaluate what to teach and how to teach it. It is clear that computer literacy is a process, not an event.

One framework for teacher education that provides a way to think about change of this magnitude is the Concerns-Based Adoption Model (CBAM) developed by the Research and Development Center for Teacher Education, University of Texas at Austin. Rooted in both theory and practice, CBAM includes strategies that may be used to guide teacher education as well as to provide a basis for determining when elements of curriculum have been implemented.

Recognition of the needs to train teachers about programming and to educate them to make judgments about how to use computers is a result of using CBAM techniques in MCPS. The first component of CBAM is Stages of Concern, a dimension of the model that links change to individuals: programming is a concern of teachers; the need for appropriate use of computers within various subject areas is a concern of administrators. These two concerns must be addressed together as teacher training and curriculum development proceed. CBAM provides a strategy for identifying where people are and what must be done to move toward a common understanding of goals. The model assumes that change starts with individuals and spreads out to the level of institutional implementation of curriculum.

In the process of moving toward a common understanding of what is to be taught to students, a consensus about what is possible is reached. For a curriculum to be useful, it must be teachable, an observation at the heart of effective teacher training and curriculum development. The connection between these two elements of program management must be monitored carefully. Failure to detect gaps in teacher training results in a general breakdown of curriculum control. Teacher training works as both probe and tool to promote change: a probe as it provides feedback to curriculum developers about what is possible in the classroom; a tool as it provides teachers with the skills, knowledge, and attitudes necessary to effect curriculum change.

THE PROCESS: Development of the course

Development of Computer Literacy for K-6 Teachers required that computer literacy be defined. Reading and consultation led to defining a computer literate person as one who has the computer-related skills, knowledge, and attitudes necessary to survive and thrive in our information-based, computer-dependent society. In this definition, computer literacy includes the functional skills of procedural thinking, using a computer and programming; the general knowledge of computer fundamentals, applications, and impact on society; and the attitudes of confidence in one's ability to control the computer, enjoyment in using it, and respect for its usefulness. The three C's of computer literacy are control, concepts, and comfort. Individuals require different degrees of these elements according to characteristics such as age, work, and interests.

Course developers next decided on objectives, an organizational structure, and a course outline. Five objectives state that upon successful completion of the course, participants will be able to

1. identify and describe the major components of modern computer systems and how they differ from earlier computing devices;
2. relate the logic of elementary computer programs to thinking and problem-solving skills taught in the regular K-6 curricula;
3. identify and use current resource materials to support student learning about computers and the many ways that they may be used;
4. design and/or select student learning activities that focus on the development of basic skills;
5. implement general computer literacy activities and identify ways to provide for diverse student needs and interests.

The course was organized as 15 three-hour sessions to be offered one evening a week in the MCPS Microcomputer Resource Center. Each session was designed to include a general computer literacy activity and a hands-on lab with the Commodore PET. The activity consists of group participation projects which can be adapted for teachers' classrooms; visits to MCPS computer sites to view central computer operations, word processing, and information retrieval; videotapes, filmstrips, and speakers on various aspects of computer fundamentals, use, and impact. The lab consists of a lecture/demonstration on BASIC programming and an hour of using the PET. Participants work in pairs using a text written by MCPS for gifted students. A class consists of about 30 participants and two instructors.

The next task was to identify and obtain instructional materials and to write agendas, activities, and handouts for each session. Newspaper and magazine articles were collected for research and classification activities. Lab sheets referenced to specific pages in various basic programming texts were written for each session.

Assignments were designed to meet State of Maryland requirements for granting in-service credit. Assignments to help teachers obtain materials for activities include collecting newspaper/magazine articles in a newsfile, collecting computer input/output and other products in a scavenger search, and identifying computer-related materials in the teacher's own school.

The major assignment is a self-designed project to be used in the participant's classroom. This project may be a computer program, a mini-unit, a center, an individualized learning packet, or other approved plans on any appropriate aspect of general or functional computer literacy. Each project must be based on student objectives from an approved list, either the MCPS gifted objectives or those in the National Science Foundation funded MCPS/HumRRO (Human Resources Research Organization) computer literacy guide.

Content and organization of the course are based on needs of the school system and the teachers and on a carefully considered definition of computer literacy. Teaching strategies are based on concepts from the CBAM model as well as standard teaching practices. CBAM teaches that change is a highly personal experience and that because change is brough about by individuals, their personal satisfactions, frustrations, concerns, motivations, and perceptions are of great importance. To identify and respond to the concerns of individuals as well as of the group, developers devised a system of communication involving a Work Record form which participants fill out at the end of each session. The instructors read these weekly comments, write replies, and provide materials/information or adjust agendas as needed.

During lab time, participants all take part in the lecture/demonstration but proceed at their own pace through practice exercises. Instructors circulate during lab, work with individuals, and try to differentiate instruction as needed.

At all times both instruction and learning activities are delivered in such a way as to encourage modeling by the participants when they return to their own school situation. The modeling idea is fundamental to the course because not only do teachers need to know the content, but they must be able to deliver it to students. Depending on the teachers' competency and comfort level, they can directly model their lessons after the class or use the content and materials in their own way.

THE PRODUCT: Experience of the first year

An essential part of the course is evaluation, which occurs weekly. An extensive end-of -semester review is performed. The evaluation form contains three parts: Part I asks participants to rate items in the categories of instructors, instructional activities, materials/equipment, and assignments on a scale of one, low, to five, high, and to add any written comments. At the end of spring, 1981, the overall class rating was 4.86. A major concern of the participants was that they be allowed to have a computer in their own classrooms. Many asked for more hands-on lab time.

Part II is a growth scale asking participants to rate their knowledge/skill level at the beginning of the course and at the end for items such as machine fundamentals, supplying data, branching and looping, subroutines, graphics, flowcharts, history, hardware/software, applications, and impact. Self-rated growth at the end of spring, 1981, revealed an operational understanding of all topics with many participants feeling that they could teach most of them.

In Part III participants are asked to respond to questions about the strengths of the course, suggestions for change, the use of the course in their teaching assignment. Comments from spring, 1981, participants testify to what can be accomplished in one semester of intensive work:

> Best course I ever took! I want to be included in the class that is to follow.
>
> A super course! I'm totally addicted to computers and will do all in my power to get our school "onto" computers.
>
> I found the course very exciting and fun, and I learned lots. I am computer literate!

According to the CBAM model, the best testimony to the success of the course in bringing about change is that of the teachers who went back to their schools, persuaded principals that they needed microcomputers, and are now using them successfully with students in their own instructional program--with primary, upper level, gifted, and special education students. All students, as well as all teachers, with training can use computers!

FACTORS RELATING TO THE UTILIZATION OF INFORMATION BY SUPERINTENDENTS FOR DECISION-MAKING

Brenda Somers

Divisional Assistant, St. Louis Public Schools

INTRODUCTION

With continued financial uncertainty, most school districts across the country are being forced to re-examine their priorities and to assess their efforts and objectives. This increasing pressure for accountability has made superintendents acutely aware of the public scrutiny of their decisions. It is axiomatic of the decision-making process that relevant, reliable and timely information is essential. Information is necessary for the major steps of the problem-solving process: identifying the problem, generating alternative solutions, selecting a solution, and implementation.

STATEMENT OF THE PROBLEM

The purpose of this study was to examine some personal and situational characteristics of superintendents as the characteristics relate to information utilization for decision-making. First of all, do superintendents exhibit substantially different levels of information use for decision-making? If so, what factors discriminate between high and low utilization of information? Does the size of the school district (enrollment) distinguish between high and low utilization? Other factors have been studied in business and industry as effecting information utilization.

Lucas (1978) found reasonable support in business and industry that individuals with differing decision styles have differing levels of use of an information system and different actions. Some of the personal factors leading to differing levels of use of information were explored in this study as independent variables. Included in this study were five independent variables in addition to size of organization. They were years in the superintendency, highest degree attained, the recency of the degree, decision-making style or type, and the number of management information system (MIS) workshops or seminars attended in the last ten years.

Little research has been conducted in this area in education. Aragona (1976) attempted to discover whether the per pupil expenditure of a district related to the utilization of information by superintendents in New York state. This study used a modified version of Aragona's instrument for collecting a score of information utilization as the dependent variable.

IMPORTANCE OF THE STUDY

This study has been conducted to contribute, in an exploratory fashion, to the body of knowledge of educational leadership. In these times of rising costs, demands for accountability, and declining enrollments, the decision-making skills at the level of the superintendency come into sharp focus. Concurrently, while the decision-making process requires accurate and pertinent information, an information explosion has created a "future shock" confusion by overloading managers at all levels.

As the management of public school systems becomes more complex, it seemed necessary to examine the basis for decision-making of the public school superintendent. Since the literature points to the premise that the utilization of relevant and reliable information is essential for improved decision-making, it was desirable to evaluate the superintendents' perceptions of the utility and value of information in decision-making.

Furthermore, there was a need to study the variables that distinguish between those superintendents who value and utilize information in decision-making and those who do not. While the entire universe of possible variables which may distinguish among information utilizers and non-utilizers could not be studied, those personal factors found to be significant in the utilization of information outside the field of education needed to be studied. In this way, generalizations made about managers in business and industry may or may not be applied to superintendents of public school systems.

Finally, there was a need to study the variables of the superintendents' academic training and experience as it relates to high and low utilization of information for decision-making. Such findings may generate recommendations that new or additional emphasis be given to decision-making, information utilization and information processing theory in the academic preparation or continuing education of superintendents.

METHODOLOGY

A sample survey research method was was employed. The public school districts of Missouri with grades K-12 or 1-12 were placed in rank order by size or enrollment of district and stratified based on National Center for Educational Statistics categories. Either the entire stratum was selected or a systematic sample was selected, depending upon the number of districts in the stratum (stratum >60 were sampled). Mailings were made to 297 of the 454 Missouri public school superintendents. An original response rate of 74.4 was obtained, with 70.0 percent usable.

Superintendents were asked to provide on a matrix the perceived value of 16 data elements in the four subgroups of costs, programs, achievement, and media in their major decision-making areas. The decision-making areas were: planning budget, planning change, organizing community support/political, appraising comparability fairness, appraising quality indicators, and appraising administrative leadership. In addition to this Information by Functional Utilization Matrix, (IFUM), superintendents were asked to respond to the Information User Characteristic questionnaire which asked for the independent variables: number of years in the superintendency, number of MIS workshops/seminars attended in the last 10 years, and decision-making type. The decision-making types included the following three:

a) Most often Analytical: quantitative, systematic, using data and logic.

b) Most often Intuitive: heuristic, using perception and little data.

c) 50% of the time Analytical and 50% of the time Intuitive.

Before frequency data were scrutinized, the independent variables were grouped where necessary, that is, for years in the position of superintendency and dates of highest degree (which in all cases was the most recent degree). The question regarding the number of MIS workshops/seminars attended was deleted from the study because 25 percent of the respondents were considered not valid.

After total scores on the IFUM were obtained for a dependent variable, an analysis of variance was performed on district size. It had been determined a priori that if no significant differences existed, the design would be collapsed to disregard district size. If significant differences were found, the size category would be continued as an independent variable, thus controlling for a nuisance variable and answering a research question.

The following were the remaining stated hypotheses:

1. Superintendents with the least number of years in the same role will score higher on the IFUM than

those with the greatest amount of tenure.

2. Superintendents with the most recently earned degrees will score higher on the IFUM than those with less recently earned degrees.

3. The type of graduate degree that superintendents have attained will significantly distinguish high or low scores on the IFUM.

4. Superintendents who perceive themselves as analytical decision-makers will score significantly higher on the IFUM than those who perceive themselves as intuitive decision-makers.

5. Superintendents who have attended management information seminars or workshops in the last ten (10) years will score significantly higher on the IFUM than those who have not.

Various subset scores on the Information by Functional Utilization (IFUM) were also used to determine whether any of the personal characteristics show significant differences on a subtest basis, as opposed to a total score. The above given hypotheses could be used, substituting the scores on one of the four subtests--Costs, Program, Achievement, and Media--for the total score on the IFUM. One-way analysis of variance was used to test the abofe hypotheses.

The final research question was: which personal characteristic best distinguishes between superintendents with high and low scores on the IFUM? Discriminant function analyses was used, both direct and step-wise, to determine the answer to this research question.

SUMMARY OF THE FINDINGS

Using frequency data the respondents can be described. Most of the superintendents (22.6%) served more than 16 years as superintendent as opposed to any other grouping of years, although the mean number of years in the superintendency was 9.65. Most respondents (39.4%) had earned a Specialist's degree as the highest earned degree than any other degree. The years the highest degrees were earned ranged from 1981 to 1947, with an average of 1970. Over 51.2 percent of the respondents considered themselves to be most often Analytical in their decision-making rather than Intuitive most of the time or Intuitive half of the time and Analytical the other half. The respondents reported that they attended anywhere from zero to 50 Management/Information System (MIS) workshops or seminars in the past ten years.

The statistical analysis used to test the stated hypotheses and to answer the research questions yielded several results. First of all, it was found that there were differences in utilization of information for decision-making among superintendents as measured on the Information by Functional Utilization Matrix. This was determined by merely comparing the 208 total scores, which ranged from a low of 149 to a high of 433. It was from the basis of this finding that further analyses could be conducted.

The size of the district or the enrollment of the district did not have a significant effect on the dependent variable, total score on the IFUM. Therefore, it was not deemed necessary to continue the use of size as an independent variable to control a potential nuisance variable. As well as conducting analysis of variance tests on the dependent variable, total score on the IFUM, analysis of variance was run on the four subgroup scores of Costs, Programs, Achievement, and Media. It was found that statistically significant differences existed only between the respondent's district size and the Media subgroup scores, not the Costs, Programs or Achievement subgroup scores, nor the total score.

Further analysis yielded no significant differences on the continuous dependent variable on the IFUM using the number of years in the superintendency as the independent variable. Therefore, it was found that superintendents with the least number of years in the same role did not score higher on the IFUM than those with the greatest amount of tenure, as hypothesized in the first research hypothesis.

Similarly, no significant differences existed among total scores on the IFUM based on the recency of a completed degree program. Using analysis of variance again, it was determined that superintendents with the most recently earned degrees did not score higher on the IFUM than those with less recently earned degrees as was proposed in research hypothesis two.

In addition, superintendents who perceived themselves as Analytical decision-makers did not score higher on the IFUM than those who perceived themselves as Intuitive decision-makers, as proposed in research hypothesis three. No significant differences less than or equal to the .05 level were obtained by way of the analysis of variance test.

An additional analysis was conducted by converting the continuous dependent variable to a categorical variable and by eliminating the middle third of the group to discriminate between high and low scores on the IFUM. The same independent variables of years in the superintendency, highest degree attained, recency of highest degree, and decision-making type were used as the discriminating variables in discriminant function analyses. Only the years in the superintendency as a discriminating variable was found to be statistically significant at the $P \leq .05$ level for discriminating between low and high scoring groups.

IMPLICATIONS FOR EDUCATORS

What do these findings say about the decision-making leaders of our educational systems? If neither the level of higher education, the type of education, nor the recency of that advanced education have a significant effect on the utilization of information for decision-making, the educators of educational administrators need to evaluate their programs. The current programs appear to have little effect on the value a superintendent places on information utilization for decision-making. Furthermore, if the number of years one spends in the role of superintendent most significantly determines the possibility of becoming a low or high utilizer of information, what is a superintendent to do to improve his or her decision-making ability? If one believes in the education process as a change agent, one must look to the education of the superintendents to affect change in decision-making ability.

Therefore, the strongest, most obvious implication that can be made from this study is that the programs undertaken by potential superintendents or undertaken as continuing education by superintendents should be examined. The programs and the included course-work designed, developed and implemented to train school administrators could be more effective and efficient training for quantitative and qualitative decision-making and more encouraging in the use of information processing as a valid step in sound decision-making.

BIBLIOGRAPHY

Aragona, L. The functional utilization of information for decision-making by chief school officers. Unpublished doctoral dissertation, Teachers College, Columbia University, 1976.

Lucas, H.C., Jr. Information systems concepts for management. New York: McGraw-Hill, 1978.

APPLYING INSTRUCTIONAL AND COMPUTER TECHNOLOGY TO THE DEVELOPMENT OF IEP's: PROJECT RECIPE

by

George A. Reid, Jr.
and
H. Sanders Bell, III

2418 Hatton Street
Sarasota, Florida
33577

Phone (813) 953-5000

THE INSTRUCTIONAL SYSTEM

Project RECIPE (Research Exchange for Computerized Individualized Programs of Education), was designed to attack the problems of poor performance on the Florida State Student Assessment Test and very high nonpromotion rates for Specific Learning Disabled (SLD) students in the elementary grades (K-6). Instructional technology was sought out to provide a solution. A part of this instructional technology was employed to look at the problem comprehensively in terms of I.E.P. (Individualized Education Plan) development and the relationship of achievement of goals and objectives designated on the IEP and student progression. Further analysis of the problems of depressed State Assessment performance and low progression indicated the need for a multifaceted instructional system combining management and instructional techniques for the resource room teacher. This system would support the special education teacher as a resource through development, implementation, and evaluation of the IEP objectives selected for a given student.

The solution design required some degree of standardization of objectives, tests, and management processes within the targeted group of teachers. The section to follow will describe the instructional design model employed in the design of RECIPE products and procedures.

1. Initially, to design an instructional system as a resource for teaching, one must get a clear consensus of the rationale or goals of the existing program. The rationale should be student-based and clearly show the reason for the existence of the program (i.e., its relationship to the mainstream curricular program as well as the distinctive features of the program involved).

2. From the rationale, student-based goals or program-level objectives should be developed. The objectives should be of a nature that will communicate to parents and the community and to regular education personnel the overall aim of the program.

3. Program-level objectives must then be analyzed to produce the specific instructional objectives that they encompass. Task analysis is the most appropriate method to use for analysis of "intellectual skills." Other methods can be used for analysis of motor skills, social/attitudinal skills, or memorization or procedural tasks. Intellectual skills broadly include the instructional areas of mathematics, language, reading, writing and other academic or academically related skills.

4. An assessment system must then be developed for pretesting and posttesting each objective listed in the previous step. (Test items need not be written if they can be found elsewhere and are deemed valid and reliable).

5. Finally, one must design, develop, and field test instructional resources (student and teacher materials) to aid the teacher in instructing objectives of the system.

The steps listed above, when accomplished, provide the basis for an instructional system geared toward the individual student. Beyond the instructional components, the computer can provide a great degree of assistance with management, record-keeping, reporting, and administrative monitoring of such a system.

Implementing an Instructional Management System for IEP's

The following procedures provide a generic outline for implementation of an instructional management system with students involved in the IEP process.

I. A child's eligibility for a special program is determined in part by his meeting criteria as determined by established test instruments. The results of diagnostic and eligibility testing, along with information from the parent, the classroom teacher, criterion-referenced testing, initial observation, and other sources are used to establish the child's needs. The child's needs are best expressed as discrepancy statements (i.e., the child cannot do something that he should be able to do).

II. This second step requires that the child's "present level of performance" be determined with respect to the area of need.

III. If present level of performance was stated as speific skills that the child can or cannot do, then a hierarchical sequence chart gives the IEP developer a selection of logical objectives to be taught next. These objectives would be proposed for the IEP and would be stepping stones toward eventual accomplishment of overall goals set by IEP planners on the basis of the student's ability to achieve in his need areas.

THE RECIPE MANAGEMENT SYSTEM

The RECIPE Instructional Management System in Sarasota County is aided by a main-frame computer (NCR 8555 model). Each elementary school has a printer terminal tied to the main computer via phone lines (LA 36 remote terminal). With the printer terminal is an optical scanner which reads pencil marked forms for reporting student progress. Many functions are handled by the SLD teacher communicating directly with the computer via the terminals. Other functions are handled by "batch" processing, where the teacher sends written information to be scanned or keypunched, and receives reports, etc., through a courier system.

To assist other school districts with adoption of the computerized RECIPE system, the main-frame system is being redesigned and programmed for the Radio Shack Model II microcomputer. Field testing and debugging of this new system is currently in progress in Sarasota County.

COMPUTER-BASED MANAGEMENT SYSTEM

I. DEMOGRAPHIC INFORMATION BASE (Optional Student Base)

1. STUDENT INFORMATION
    - Demographic data
    - Previous Placement
    - Health/Medical Information
    - Referral Information
2. ELIGIBILITY AND RECOMMENDATION - STAFFING
    - Permission/Consent Dates, Names
    - Eligibility for Program
    - Program Placement
3. INDIVIDUAL AND GROUP TEST RESULTS
    - Diagnostic, Eligibility Tests listed by Test, result, date, and administration.

II. IEP AND IMPLEMENTATION PLAN

1. GOALS AND OBJECTIVES WITH EVALUATION CRITERIA
    A. Goals and short term objectives for Annual IEP compliance document
        - Selected by IEP Committee/updated yearly or upon request
2. QUARTERLY IMPLEMENTATION PLAN (Instructional Objectives)
    A. Selected by the Resource Room Teacher
    B. Reported on by Optical Scanning at Remote Terminal
    C. Generates Quarterly Report to Parents
    D. Generates Mastery Labels for Students
3. QUARTERLY REPORT TO PARENTS, CLASSROOM TEACHERS, EXCEPTIONAL STUDENT RECORDS
    - uses mastery codes to indicate student progress
4. CURRENT IEP PROGRESS REPORT
    - requested by teacher on-line, reports current status of all IEP objectives
5. STUDENT MAP PROGRESS
    - allows teacher to probe certain objective listings (on-line)
6. STRATEGY SELECTION
    - prints strategies and activities for selected instructional objectives
7. MASTERY UPDATE REPORTING
    - on-line reporting of objective mastery by teacher (optional)
8. OPTICAL-SCAN PROGRESS REPORTING
    - on-line quarterly scanning procedure

343

9. SCHEDULING AND GROUPING REPORT
   - groups students by objectives selected for instruction

III. ADMINISTRATIVE MONITORING REPORTS

1. Summarized IEP Progress by Program, by School, by Teacher
2. Categorical Listing of Content Areas Appearing on IEP's
3. Summaries of Demographic Data by Program, by School, by Teacher
4. Listing of Computer Transactions, by School, by Teacher

Project RECIPE was validated by Title IV-C in August 1981. The system is presently in use in all elementary school SLD resource rooms in Sarasota County, Florida, and is available for dissemination throughout the State of Florida.

STUDENT SELF-TRACKING USING A MICROCOMPUTER

Mary Winifred Strong

Principal, Monongalia County Schools
Morgantown, West Virginia

Introduction

The computer has been used in the field of education for data management as well as for instructional purposes for the last ten years. Generally, the teacher has made use of the computer's quick ability to collect student data to store it, to retrieve it, to manipulate data, and to prescribe or diagnose the individual student's program.

In a report to school superintendents in 1974, Robert Scanlon indicated that the actual usefulness of computer managed instructional programs was poorly documented. Some information was not available on the programs used in various school systems. Local school systems were developing programs instead of major instructional systems and there was no data available on the cost-effectiveness of computer managed instruction (Scanlon, 1974).

Secondary schools, colleges, and universities have utilized computers for instruction, especially in the subject areas of math and science. Elementary school children have been exposed to computer managed instruction in reading since 1972 (Lorenz, 1976). The Wisconsin Design System for reading was used by teachers in the elementary schools to diagnose and prescribe pupil needs through more efficient collection and storage of student and program information. Mathematics and science subject areas were later added to the computer managed instructional design of the Wisconsin System for elementary students. This has also been the trend in other parts of the United States.

Education has been somewhat behind industry and business in the development of computer applications, but is now beginning to realize its potential.

Project Description

The advent of the microcomputer into the commercial market has made it possible for small organizations to have the same types of programs as large companies with main-frame computers. This is due to modern technology and the fact that microcomputers are much less expensive than computers. The Title IV-C project described in the following discussion delineates some of the previously mentioned educational applications of the microcomputer in a small elementary school.

The purpose of this study was to determine the incentive effect of data self-recording by elementary children using a microcomputer upon reading achievement, sex, and grade level. Attitude toward reading was also assessed during the course of the project.

The microcomputer model described in this paper was a direct result of a Title IV-C grant issued by the West Virginia State Department of Education. The grant, called "Project S.U.M.M.," was implemented in a small elementary school in Morgantown, West Virginia (Strong, 1979).

The initial funding for the Title IV-C grant was $7,000.00. The TRS-80 Level II microcomputer which was originally purchased for the project cost approximately $3,000.00 and the printer was an additional $1,000.00. The remaining grant money was allocated for teacher in-service, training, service contracting, software development, travel, and project evaluation.

The three treatment periods conducted during the first year of the grant were: (1) no points issued for correct reading performance and incentives given; (2) aide recording of points earned for correct reading performances and incentives given; and (3) student self-recording on the microcomputer of points earned for correct reading performance plus incentives given. Each of the three treatment periods lasted for a duration of 29 days.

The 57 intermediate students involved in this experimental research daily attended a reading laboratory where an individualized reading program was housed. Every pupil used the microcomputer during the third treatment period to keep a record of their reading achievement. A "take-home" print-out in graphic form of daily progress was issued to each student at the end of the week.

Results

The results of this study did not indicate that there was more achievement or work produced during the third treatment period--the period in which the microcomputer was used for student self-tracking. The second treatment period during which the aide recorded the reading work produced was the highest achievement period followed by the first period (no treatment). The third period (student self-tracking) was the lowest in achievement. This data was treated by using a 2 x 3 x 3 factorial analysis of variance. It should be noted that the independent variable of treatment period was statistically significant ($p < .001$). Perhaps, if this study were replicated, the order of the treatment periods could be rearranged and different results would occur (Strong, 1981).

Sex and grade level did not have any significant effect upon the reading work performance of the students in the study or upon their enthusiasm in using the microcomputer to record their achievement. These data were also treated by using a 2 x 2 x 3 analysis of variance with repeated measures.

As an assessment of attitude, the students who participated in this study responded to an objective questionnaire pertaining to their attitude toward going to the reading laboratory during all three treatment periods. The results of the questionnaire showed that more students indicated a positive attitude toward going to the reading laboratory in treatments two and three than during treatment one. A chi square test was applied to the frequency of student responses recorded on the questionnaire. The results indicated that the attitudinal responses were significant beyond the .05 level ($X^2 = 18.8$). Thus, the pupils seemed to enjoy recording their achievement using the microcomputer.

The microcomputer as employed in this project did not have a significant effect upon student reading achievement per se. However, student attitude toward going to the reading laboratory was effected positively using the microcomputer for self-tracking of data.

The pupils using the microcomputer in this study could actually plot their progress and thus become aware of what was needed to improve achievement and goals. Throughout this immediate feedback system, reinforcement of goals were maximized and the microcomputer was used as a secondary incentive device. The attitude toward going to the reading laboratory showed improvement. It would seem, then, that using the microcomputer to plot progress was an incentive that did work in improving attitude.

Lipe and Jung (1971), in a review of research in the area of educational psychology, conclude that response incentives which are called self-managed may appear to be the same as incentives per se. However, the idea of self-determination, set goals, and

352

self-management of rewards are long range objectives of most educational programs. They are also a step toward self-motivation--the highest educational goal and the key to student learning and academic achievement.

Bibliography

Lipe, D. & Jung, S. M. Manipulating incentives to enhance school learning. Review of Educational Research, 1971, 41, 249-280.

Lorenz, T. B. MICA-managed instruction with computer assistance: Level five. ERIC Document Reproduction Service, ED 122 285 988, 1976.

Scanlon, R. C. Computer managed instruction and future directions. ERIC Document Reproduction Service, ED 160 072, March, 1974.

Strong, M. W. Project S.U.M.M. Title IV-C Grant. West Virginia State Department of Education, July, 1979. DEV 3103.

Strong, M. W. Student self-tracking with a microcomputer. Unpublished doctoral dissertation, West Virginia University, 1981.

BASICS, NOT BASIC: OR TEACHING WRITING TO THE COMPUTER

Dr. Ann Stuart

Professor of English

Software documentation is writing which means that its nature is to inform. To perform that task successfully, the basic principles of rhetoric, applicable to all good writing, are useful tools to the writer.

In 1980 I developed a Program of Technical Writing for the School of Engineering at the University of Evansville. That program includes courses in scientific and laboratory report writing, proposals and project designs, and software documentation. While I worked to develop the curricula and to determine teaching methods, I discovered that the basics of writing and the means of teaching writing are similar regardless of the group of people trying to learn. In the following paper, I intend to demonstrate how this recognition influences my teaching of software documentation.

One quality of good expository writing is efficiency. To introduce the question of whether a piece of writing is effective, I often begin with a simple memo like this one:

M-E-M-O-R-A-N-D-U-M
December 7, 1981

TO: Manfred Schauss
Carolyn Holland
All Programming & Operations Personnel
FR: Lynne Pike
RE: S222P

There is now a job steam available to access data element 016 in the alumni file. The job number is S222P, and it can be found in the file cabinet by the stock room door.

Previously, only the alumni's employer code number in 016 was accessable. By using the S222P job stream, the actual name of the company can be obtained.

Although S222P requires a longer execution time, it will save at least ten times this amount of time for the Alumni Department. They will no longer have to manually look up the company name according to the employer code number.

This new job stream will be implemented December 9, 1981 after a briefing in my office at 1 p.m. Please contact me if you can not attend.

****

Reasons why this piece of writing is ineffective include individuals not being identified, vague titling of subject, improper use of the paragraph to indicate separation of ideas, failure to include information for the receiver to contact the sender, and poor use of chronology to inform the reader of what was available and what is now available and how the new is better. Spelling and grammatical mistakes are also noted because one is not going to be thought effective if one cannot communicate accurately and correctly.

What one teaches in order to produce a better memo is familiar to any English teacher: organization, logical development, and completeness. The matter is the same; the basics of good writing apply to all types of writing.

Determining a practical form for the function of one's writing is another basic principle applicable to software documentation. This simple set of instructions illustrates the point:

OPERATION INSTRUCTION FOR A101

This job stream is used to select married alumni from the alumni file.

1) Mount the REMOVE4 disk pack on disk drive 200.
2) Mount a large scratch tape on tape drive 181.
3) Run the A101 job deck into the reader.
4) When the console asks for the date, type the current date in the following format:

   / / DATE, 05/05/81
5) After the job has ended, BG WAITING FOR WORK will be displayed on the console. Remove the scratch tape and the REMOVE4 disk pack.
6) Print the output on Fanfold paper. Lay the printout and job deck on the output table.

*If the job cancels type in MSG BG and have the system programmer look at why it cancelled.

****

The form of this writing is inefficient partly because the writer did not determine the need for a distinct list of commands for the actions to be performed. Items 1,2,&3 begin properly enough with commands; however, item 4 begins with an introductory phrase as does item 5; moreover, item 5 contains two commands--wait and remove--where one command has been the previous pattern. Finally, the comment placed at the end of the directions is offered too late to be effective.

By applying parallel structure to the set of commands and by developing a form that allows comment to be separate from instruction yet placed relevantly in the text, this "operation instruction" material becomes efficient.

The point of these and the other examples I present in my presentation is that one can offer effective criticism to the writer of different kinds of software documentation in the language traditionally associated with composition. While the basics of rhetoric do not address all of the needs of the writer of software documentation, they provide a solid foundation to begin effective, well organized material.

ESTABLISHING A HARD-DISK BASED MICROCOMPUTER NETWORK

Jack T. Tinsley

Director, Computer Center
University of North Florida

Al Hodges

Computer Center
University of North Florida

A little over a year ago we at the University of North Florida began thinking about microcomputers and questioning ourselves about the potential of microcomputing in education. We were quite skeptical -- or maybe I should say, I was very skeptical about the value of microcomputers. In fact, my first impression (and it lasted for sometime) was that micros were "nice toys" but not of much practical use. And, quite frankly, it was at the Florida Instructional Computing Conference in February of 1981 that my mind was changed as to the applicability of micros in the classroom.

There were several of us from the University of North Florida who attended last year's Florida Instructional Computing Conference - some from the Computer Center and some faculty members. When we got back to campus we were convinced that action had to be initiated to get our university more in tune with the needs of the educational community as it relates to computing.

It was recognized that teachers were and are being asked to use microcomputers in the classroom. We were not exposing our students (some of which will be our future teachers) to computer technology. Along these same lines the teachers were being asked to evaluate CAI software or use CAI software - and, as you know, there is some good software available and some not so good. The universities need to expose students to CAI software and provide the tools required to evaluate software as to its place in a given course.

Another discovery was that some faculty members were already acquiring microcomputers. There were/are TRS 80's, APPLES, TERAKS and NORTHSTAR HORIZONS, to name a few. It became apparent that some direction with regards to microcomputing needed to be established. To address this proliferation of microcomputing the University Computer Users Committee was called upon. It was determined by the Committee that a particular microcomputer would be the standard and that all purchases of computing hardware and software would be approved by the Committee prior to acquisition. It was also recognized that as a learning institution there would be times when deviation from the standard microcomputer would be allowed, still subject to the Committee's approval.

The intent with regard to a set of standards is to obtain like hardware to accommodate interchange of peripherals, and we plan to establish a "bank" of parts so that we can maintain the equipment in operating order. Another reason for standardizing hardware is to obtain software that could be used by more than one individual. The Computer Center coordinates the acquisition of hardware and software. In this way an inventory is more easily maintained and we avoid duplicate purchases of software.

We chose to network 18 Apple II microcomputers into a 20MB hard-disk system. The primary reason for this choice was to more effectively utilize the micro lab as a classroom. Our thinking was it would take less time for all students to load the same program from a hard-disk system than to pass around a floppy disk. By the time you pass a disk around to 18 people you've lost 20-25 minutes of class time. What we failed to consider was that the software you buy that is copyrighted cannot be copied to the hard-disk system.

This is where we need to get the software vendors to work with us to establish a mechanism to accomplish this task. We are very aware of the copyright laws and we do not want nor do we intend to violate these laws. But to effectively utilize a hard-disk system we must work toward providing the capability of having software on hard-disk.

We have contacted some of the vendors with regard to this problem and they seem willing to work with us. One vendor has provided us with unprotected copies of software we purchased from them so we could copy it to the hard-disk. The person we have in charge of our microcomputer activities has developed a method so that the software cannot be copied off the hard-disk by anyone using the software.

It was recognized by the Computer Users Committee that to have an effective program utilizing micros it is imperative that a person, with microcomputer hardware and software expertise, be hired to manage the facilities. I would like to emphasize this point. If you want to have a successful program you must be willing to dedicate some personnel resources to this task.

To give you a better understanding of "why" I'll point out one of the mistakes we made.

We purchased 8 copies of PASCAL and the associated memory cards. With the proper expertise we would have purchased only one PASCAL language and we could have bought additional memory for all 18 micros for what it cost us to buy the 8 PASCAL cards.

Another key requirement when hiring a person to manage the micro facility is the individual's communication skills. When dealing with such a diverse group of users and potential users the person must be able to effectively communicate.

We have a job of educating our faculty about the use of microcomputer. Another task this person performs is regularly scheduled seminars on general use of the computers, and we have and will continue to offer specialized seminars on the use of software like VISICALC and GENIS I (a courseware development software package).

To try and head off problems before they arise we ask all faculty that will be teaching a course which utilized the microlab to provide us with time to introduce the students to the computing facility. By familiarizing students with the correct way to turn the computers on and the way to handle the floppy diskette drives (when needed) we believe we will avoid damage to the equipment.

We are also compiling a file of software that is purchased. Periodically, we send a copy of the software list around to the faculty for their information. This list is also used to help us avoid buying the same piece of software more than once.

I believe you can see from this short account that a person dedicated to managing the micro facility is vital. I would suggest that if you are not prepared to dedicate the personnel resources necessary to establish and maintain a microcomputer network, then don't attempt it.

When preparing a RFP (Request for Proposals) we suggest that you require, where possible, local service because you may rest assured you will have hardware problems regardless of the computer selected. The important task when it happens is getting the machine repaired as quickly as possible. As more and more courses are developed around the use of computers then they must be available. A local vendor will be far more responsive to your needs than one located several hundred miles from your location.

The Hardware Configuration

The University of North Florida has eighteen Apple II+ computers interconnected to two 20-megabyte Corvus hard-disk drives in a "constellation" configuration so that simultaneously each computer appears to have exclusive use of one 20-megabyte disk drive. These Apples are located in a separate microcomputer lab adjacent to the university's main computer terminal room, and it is managed by the UNF Computer Services staff. It is available for use by all university students and faculty.

Introductory seminars have been presented by the Computer Services staff to familiarize the faculty with the Apple computers and to explain exactly how each faculty member can make best use of the lab. Tutorials on VISICALC, Bell and Howell's GENIS I Courseware Development System and other software packages have been given for faculty members. Follow-up sessions on these and other topics will be conducted as needed so that the faculty can utilize the microcomputer lab effectively.

Students using the language BASIC in any course will be required to use the Apple microlab. This will allow the University of North Florida's mainframe systems to operate much more efficiently. Students will be able to copy their program workfiles from the Corvus disk system to their own $5 floppy diskette so that all work can be saved from one session to the next. Hard copy printouts will also be readily available at the end of each session for diagnostic purposes or to hand in as a completed class assignment.

The only serious limitations we now face concerns the use of copyrighted software programs. Software vendors are being cheated out of a large percentage of their profits by individuals illegally copying their programs instead of buying them. At the present it seems that any new diskette copy protection scheme can be and will be broken and that new automatic copy programs will continue to become available. However, with the Corvus system we do not have to worry about diskette copying. We only have to control access and prevent the copying of programs onto diskettes.

At UNF we have recently utilized the common method of protecting an executing program using a keyboard lockout mechanism which prevents program interruption and thus prevents illegal copying to a diskette. This procedure is foolproof because the Corvus disk system uses ID and password schemes to allow or deny access to any of its volumes. Also, whenever any user signs on to the Corvus with a valid ID, a "HELLO" greeting program is automatically executed each and every time before the user is given any control of his computer. This special greeting program immediately locks out keyboard interruptions and allows only menu selection of the desired program. All that is necessary to continue the foolproof protection is that the selected program immediately executes the one line of code which prohibits keyboard interruptions. Thus a user is "locked in" and can only regain control of his computer by turning it off and back on again, erasing the program from his computer's memory and signing him off the Corvus system.

Only the one Corvus system manager who has access to and control of all passwords and volume access has the ability to load and unload the copyrighted programs. All other users of copyrighted programs must sign on with the special keyboard lock-up greeting program.

We are now trying to contact various software vendors and arrange for this simple protection scheme to be implemented. We feel that it is of great benefit to the vendors for thousands of students at the University of North Florida to have a chance to work with their products on our system. This will stimulate interest and sales of microcomputers and their particular software products. Most importantly to us, this will allow us to effectively utilize educational software on our microcomputer system without risking the dangers of copyright violations.

TO COMPUTERIZE OR NOT?
A QUESTION OF GIGO!!

Robert A. Zuckerman, Ed.D.
Kent State University
(Abridged 2/22/82)

Colleges contemplating the acquisition of computers need to consider that the purpose for having the computer is in terms of what the machine is supposed to do (viz., the applications) and NOT what machine to purchase (viz., the hardware). To proceed without understanding even the most basic factual information about computers can be not only disasterous but very expensive. This is consistant with the concept of "GIGO."

EDUCATIONAL USE OF COMPUTERS

The use of computers in the field of education includes drill/practice, testing, and instruction. Educators refer to CAI (Computer Assisted Instruction), CAL (Computer Assisted Learning), CATTS (Computer Assisted Teacher Training Systems), CMI (Computer Managed Instruction) and CDMS (Classroom Data Management Systems) as applications which will be greatly used in the very near future.

Drill/practice, testing, and instruction are all potential components of a CAI approach. Drill/practice activity presently is being handled by a variety of dedicated machines (viz., the hardware has the instructional program in residence and therefore only runs that program: i.e., Speak & Spell, Little Professor, Spelling B, Data Man, Digitor Drillmaster). Testing and instruction, although not necessarily on a true computer, is presently handled on dedicated teaching machines which have software (i.e., System 80). These examples are not true multi-use computers although any function performed by these dedicated machines can be done on a single computer using canned instructional programs.

Not only should the college be training teachers to use computers in the public school, but it should also employ computers in the training of teachers through a CATTS. Programs can be developed allowing the preservice teacher to interact with the computer to design instruction for target students and receive feedback on how successful the plan was, to identify off task behavior and input a management strategy/behavior and receive feedback on the effect of intervention, practice recording and interpreting assessments, and practice many other teacher skills.

More and more teachers may find that the computer is an invaluable tool for classroom data management. Teachers may take attendance by computer, keep track of lunch money, have direct access to individual student records, keep inventories of classroom materials, and have their lesson plans, unit plans and weekly plans on computer. In addition, computers offer the ability at any point in time to identify the present level of each student's educational functioning as well as the identification of what each student should learn next in the context of the curriculum on the basis of data gathered through assessments and performance during instruction. This ability would greatly facilitate and enhance the preparation of grade reports and IEP's by the teacher.

Communication with instructional computers in other classes within the school building and school district currently exist through local networks. The next step is the expansion of these educational computer networks across the state and nation. There are more and more commercial networks which schools may link up with for accessing information or for data storage when the local capability is limited and the expense of having local storage is prohibitive. [For those interested in "telecomputing" to use networks refer to the publication Kilobaud Microcomputing, Volume V, No. 10, October 1981.]

LEVELS OF "COMPUTER" USE BY EDUCATORS

Before any further discussion can take place about computers, developing a computer center, and the goals or objectives for students to achieve through such a resource center, an identification of "Levels of Computer Use" appropriate for teachers in school settings must be considered. Without an awareness of the different levels of potential use, there would be no way to sequence knowledge and skills for acquisition.

Level I

At this most basic level, the teacher uses "dedicated" instructional computers or computer-like machines. The hardware contains the program software and therefore is greatly limited in use to practice/drill activity. At the upper end of computer-like machines are the teaching machines that use special software that emulate computer programmed instruction. The greatest limit to these machines is that the software is hardware-specific (viz., the instructional programs will only run on the machine for which they were designed). To use these dedicated machines the teacher need only know how to plug them in (if necessary), turn them on or off, load software (if necessary), enter control codes or actuate switches to start the program, and how to exit or end the session.

Level II

At this level [and all levels that follow], a true microcomputer is used as the hardware and the teacher uses "canned" (i.e., commercially produced instructional programs on cassette tape, floppy disk, or plug in read-only-memory (ROM) cartridges) instructional programs. To use canned programs the teacher must have knowledge and skills to get the computer into command mode (getting machine on line and ready to receive loading instructions), load program, run program, exit program, and dismount (end session and shutting down machine).

Level III

The teacher at this level develops programs for drill or practice, instruction, and tests through knowledge and skill in using a canned, "user friendly authoring program." An authoring program calls for the teacher to have the knowledge and skills required in the previous level. In addition, the teacher must learn to use the program menus (viz., lists of options or instructions provided by the program to the user) to develop the desired instructional programs.

Level IV

This level of computer use by teachers is not an increase in the "instructional" use of the machine, but rather an additional application to support instruction and other teacher responsibilities. At this level the teacher has knowledge and skills in the use of canned user-friendly data management programs for the purpose of organizing student records, lessons, scheduling, inventorying instructional materials/supplies, preparing IEP's, preparing reports, and many other organizational or wordprocessing tasks. The knowledge and skills necessary are the same as those for Level III.

Level V

At all previous levels of use, teachers need not master even the most rudimentary knowledge and skills of programming in a computer language because this has already been accomplished for the user in canned programs. This level of use is characterized by the teacher having "necessary and sufficient" knowledge and skills in a "high level computer language" (i.e., BASIC, FORTRAN, COBOL, SNOBOL, APL, PILOT, ALGOL, PASCAL, and many others including dialects or variations of those mentioned) to write simple drill or practice and test programs. To do this level of programming, one must master a limited number of "operating system" commands (viz., to get on line and load the language interpreter) and language commands as well as commands to edit and/or store the program for future use. The most popular high level programming language for the novice is BASIC.

Level VI

At this level of use, the teacher must have greater knowledge and skills in programming in a high level language for the development of "instructional programs" (viz., programs which introduce and serve as the primary vehicle of conveying curriculum content to students). Instructional programs can be very complex through the use of extensive branching (viz., built-in program capability to skip content student knows OR provide additional assistance for content student has difficulty with as demonstrated through student responses in program) within a main program as well as the use of subroutines and data file calls and read/write operations. Instructional programs at this level of use do not include data generation or manipulation by the program. All information to be learned is resident within the program.

Level VII

The skills of the teacher who uses computers at this level includes the knowledge and skills of programming of the previous level. In addition, knowledge of mathematical models, probability, simulation and/or gaming, as well as graphics programming are required. This level of use is characterized by the the ability to develop instructional programs having simulation/gaming and/or graphics employed as the primary vehicle for learning and applying knowledge and skills.

Level VIII

This level of use will seldom be reached by an individual who would remain as a teacher in the public school. Any teacher having the skills associated with this level should be in the business of preparing canned programs for use by Level II or III users. Programming at this level requires knowledge and skills in "Assembly Language" (viz., a machine-oriented language identified as a "low level") as well as hardware operation of the CPU (Central Processing Unit, viz., the component of the computer that governs the flow of input and output to peripheral devices and memory).

INSTRUCTIONAL GOALS FOR A UNIVERSITY COMPUTER RESOURCE CENTER TO TRAIN TEACHERS

The goals established for a university computer resource center will greatly effect the hardware needs (viz., the computer and peripheral devices), software needs (viz., the canned and locally developed programs for instruction and use), and the instruction of preservice teachers. The following set of proposed goals was developed to emphasize the first four levels of computer use described in the previous section. The intent of these goals is not only to make students who will become teachers computer literate but also to assist them to develop skills to use canned, user friendly CAI and CDMS programs. [Some BASIC programming skills should be developed, but the intent of this resource center program is teacher use and not to prepare computer programmers.]

1. Awareness and recognition of the instructional use of computers in public school settings.
2. Awareness and recognition of terminology used when talking about computers.
3. Awareness of the components of a computer system as well as knowledge about what each component does within the system.
4. Knowledge and skills in getting the computer system ready for use.
5. Knowledge and skills in using the operating system of the computer to control loading and saving programs as well as file manipulation to or from peripherals.
6. Knowledge and skills to use canned instructional programs in the computer and plan for the use of such programs as part of instruction in the public school.
7. Knowledge and skill in the use of canned user-friendly authoring programs to develop custom drill/practice, test, and instruction programs.
8. Knowledge and skills in the use of canned user-friendly classroom data management system programs.
9. Knowledge and skills in programming in BASIC to develop simple "string manipulation" (viz., alpha-numeric text rather than numerical symbols) programs for drill or multiple-choice tests on computer.

[10. To be available to the college for data management and word-processing use.
11. To be used by the college for CATTS applications.]

A starting point for decision-making with respect to the acquisition of a "computer resource center" for the college most logically should begin with an examination of the current status of computer use and know-how/expertise of faculty and staff within the college as well as the university at large.

1. Identification of faculty and staff who are "computer literate" (viz., awareness of the development of computers, computer capabilities/limitations, future trends for computer use, and knowledge and skills to run canned programs or operate dedicated machines).
2. Identification of faculty and staff who own and/or use computers.
3. The applications (i.e., software programs, utilities) that faculty presently use.
4. The operating systems and programming languages that are known and used by faculty who develop programs.
5. The level of technical expertise with respect to hardware configuration, servicing, diagnosis, and optimization.

In summary, obtaining computer capabilities for training teachers may almost be as complex as using them. The expensive way to computerize is to select a machine first without considering the applications or to not truly understand what you get when you purchase a computer system. Whoever is responsible for such decisions should get to know as much about computers as possible through reading a few basic books about computers and talking with those presently using them for the same purpose. **A well-meaning but unresearched decision will result in GIGO!!!! GIGO stands for "Garbage In, Garbage Out". Translated into NON-TECH, "You get out of it, what you put into it!!"**

Consider all the information presented as simply an introduction to the task of considering the acquisition of computers for training preservice teachers in their use.

* * *

This manuscript was prepared on a H-89 computer running a word processing program and printed on an Okidata Microline 82 Dot Matrix printer.

Suggested Reading

Periodicals:

[These periodicals are listed with respect to technical nature: complex high-tech emphasis to low-tech emphasis.]

BYTE the small systems journal, McGraw-Hill Publication (Monthly)

MICROCOMPUTING, Wayne Green Publication (Monthly)

INTERFACE AGE computing for business and home, McPheters, Wolf, & Jones Publisher (Monthly)

CREATIVE COMPUTING, Creative Computing Publication (Monthly)

DESKTOP COMPUTING, Wayne Green Publication (Monthly)

PERSONAL COMPUTING, Hayden Publishing Company, Inc. (Monthly)

POPULAR COMPUTING, (Monthly)

onCOMPUTING guide to personal computing, McGraw-Hill Publication (Monthly)

Books:

Barden, William Jr., *Guidebook to Small Computers* Indianapolis:Howard W. Sams & Co., Inc., 1980.

Freiberger, Stephen and Chew, Paul Jr., *A Consumer's Guide to Personal Computing and Microcomputers* (second edition), Rochelle Park:Hayden Book Company, Inc., New Jersey, 1980.

Herbert, Frank and Barnard, Max, *Without Me You're Nothing: The Essential Guide to Home Computers*, New York:Simon and Schuster, 1981.

Osborne, Adam, *An Introduction to Microcomputers: Volume 0 The Beginner's Book*, Adam Osborne and Associates, Inc., 1977.

FOR COPIES OF THE UNABRIDGED PAPER CONTACT:

DR. ROBERT A. ZUCKERMAN
401 WHITE HALL
KENT STATE UNIVERSITY
KENT, OHIO 44242

The Design of a Structured CAI Language and CAI Support System

Marshall Harris, Department of Computer Science, University of Queensland,
Queensland, Australia

1. Introduction

My Department has 600 1st-level Computer Science students, a number which prevents us from giving the tutorial attention that is required. We therefore resolved to provide computer-based tutorials to relieve the pressure on staff and to give a better and more uniform service to the students. Lectures and a text-book would remain the primary information source.

I became convinced that the CAI systems readily available to us were not suitable, for reasons which I have discussed elsewhere (1). Instead, with the aid of a systems-programmer, I have designed a CAI system, COCKPIT (Computer-Oriented Construction Kit for Programmed Instruction and Tuition), which consists of a CAI authoring language, GASCAL (Generalised and Structured CAI Authoring Language), and a number of utilities, to be described later. The system is currently under development.

It will run on a PDP 11/34 computer, with 256KB of store, 2 5MB hard disk drives, a DH11 Asynchronous Communications interface, and 16 programmable terminals each with video monitor and 32KB of store shared between RAM and ROM.

2. Design Criteria

To provide a user-friendly system, COCKPIT facilitates all the actions and processes needed for CAI, allows the program structure to reflect the lesson structure, makes control of the lesson natural and easy, and offers easy mnemonics and syntax for GASCAL, useful and easily-used utilities, flexibility in controlling the sequencing of units (which make up a lesson or tutorial), automatic but flexible collection of statistics, and ease of top-down design of scripts.

3. Structure

3.1 Structured Programming

A COCKPIT tutorial is, of course, a computer program, although the GASCAL language is tempered to the needs of people who are not professional programmers. The techniques of Structured Programming (2)(3)(4) are now well-established as a very efficient way of designing programs. For this reason GASCAL provides all the necessary language devices for Structured Programming and top-down design of scripts.

To see how it does this, consider some typical information/instruction structures.

3.2 Unit Structures

In the COCKPIT system, a lesson consists of a number of Units, sequenced, under teacher and/or student control, by a Sequence Control System (SECS). The Unit is the major subdivision of a lesson or tutorial.

Typically (especially in subjects like Computer Science) a Unit is subdivided hierarchically, as, for example in Fig. 1.

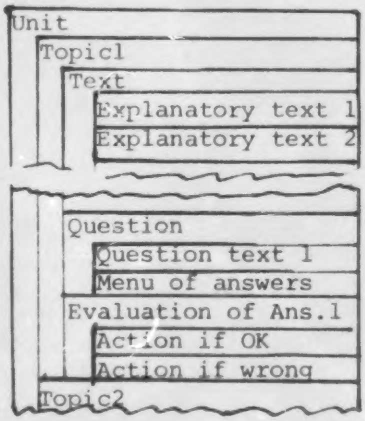


Figure 1

Hierarchical models are common, as Figs. 2 and 3, show.

In GASCAL, a Unit is written as a hierarchy of nested modules, so that Fig. 1 would be written, in outline, as Fig. 4.

```
Unit:Data Structures
  1.Introduction
  2.Abstract Data
    2.2Arrays
    2.3Queues
  3.Storage Structures
    3.1Vectors
    3.2Lists
    3.3Plexes
```

Figure 2

```
Unit:Theory of Acupuncture
  1.Overview
  2.Energies
    2.1Yang
    2.2Meridians
  3.Points
    3.1Command Points
    3.2Shu and Mu
    3.3New Points
```

Figure 3

```
Unit DATASTRUC;
  Modules
    Ext Proc TOPIC1;
    End;
    Ext Proc TOPIC2;
    .
  Commands
    Do TOPIC1;
    Do TOPIC2;
    .
End;
```

Figure 4

Fig. 4 is to be understood as follows: The Unit, called DATASTRUC, consists of several Modules, namely TOPIC1, TOPIC2, etc. (In a real script, each module would be named - "etc" can't be used! The dots represent where further modules would be written, in reality).

These modules are executed by the "Do" commands. The word "Proc" indicates that the module is a procedure, rather than a function (the distinction is explained below). The word "Ext" means that the full specification of the module is to be found externally and not within this sequence of GASCAL lines. Thus detailed specification of the action of the modules is deferred.

We can now expand the deferred modules elsewhere, as in Figs. 5 and 6.

```
Modules
  Proc TOPIC1;
    Modules
      Ext Proc TEXT1;
      End;
      Ext Proc QUESTION1;
      End;
      Ext Proc EVALANS1;
      End;
    Commands
      Do TEXT1;
      Do QUESTION1;
      Do EVALANS1;
  End; [* of TOPIC1 *]
  Proc TOPIC2;
    Modules
      Ext Proc TEXT2;
      End;
      .
    Commands
      Do TEXT2;
      .
 End; [* of TOPIC2 *]
; [* end of this sequence
```

Figure 5

```
Modules
  Proc TEXT1;
  [* output TEXT1 information *]
  .
  End; [* of TEXT1 *]
  Proc QUESTION1;
  [* output QUESTION1 text *]
  .
  End; [* of QUESTION1 *]
  .
; [* end of sequence *]
```

Figure 6

Top-down design allows a complex problem to be broken down into more easily handled sub-problems. Although GASCAL lends itself to hierarchical lesson structure, it is flexible enough to allow any kind of sequencing, such as student selection of modules.

4. The Power of GASCAL

In this limited space I can give only the "flavour" of GASCAL.

4.1 Procedures and Functions

A procedure is a segment of commands, and the variables and modules to which those commands refer. They are more powerful than BASIC subroutines, in that they can be passed parameters, to vary their actions, and may be used recursively.

A function is very similar, but it also has a value, which it computes, so that it may be used in place of a number, string, or whatever sort of value it computes.

A special "Help" procedure can be included within any module. When the student presses a certain key, the "Help Menu" is displayed, and the student can choose to skip forward or back, or simply return, or type a comment, or see what the Help-procedure offers (i.e. what text and/or questions the author has provided as help).

4.2 Ports

Some GASCAL commands allow the author to specify one or more rectangular screen areas, or "Ports", which may overlap. Text and graphics are displayed in the nominated port.

4.3 Answers

An "Answer"-command accepts the student's input, and may be part of a loop, so that the author can allow multiple tries. A time limit on answering may also be set. The number of attempts, the time taken, and whether either limit was exceeded, may be tested to control looping, and are automatically recorded on the student's Statistics file.

4.4 Answer Evaluation

GASCAL possesses powerful features for matching the student's response against acceptable answers. They include allowing or excluding redundant characters, permutation of word-order, sentence parsing, acceptance of "near-misses", etc.

4.5 Other Commands

Other commands permit testing-and-branching, reading text and graphics files, reading and writing data files, changing screen type-fonts, terminating the Unit, pausing under student control but subject to a time-limit, and showing text in any specified port.

4.6 System Modules

Many useful modules are pre-defined, including modules for manipulating character strings, and for producing screen graphics.

5. Other Components of COCKPIT

5.1 SECS - Sequence Control System

This allows the author to control the sequencing of Units. A unit may pass on to SECS accumulated information about a student's performance, which may be tested to determine the next Unit to be presented. SECS also can control how far forward or back the student may skip.

5.2 SYNDRED - Syntax-driven Prompting Screen Editor

This is for creating and editing scripts. It prompts and carries out all indentation, so the author need not worry about details of GASCAL syntax. This type of editor has been described elsewhere (5)(6).

5.3 Statistics Package

Automatically gathered student performance statistics can be processed by this package, to produce reports as needed.

5.4 Other Special Editors

Editors will be made available to allow easy production of graphics, character-sets, sound, and animation files, for use in the scripts.

References

(1) M. Harris: An Informal Assessment of Five CAI Script-Writing Systems, Australian Computer Science Communications (forthcoming).

(2) Dahl, O.J., Dijkstra, E.W., Hoare, C.A.R: Structured Programming, Academic Press, 1972.

(3) Wirth, N.: Program Development by Stepwise Refinement, Comm. ACM14,4,April 1971,pp221-227.

(4) Wirth, N.: Systematic Programming: an Introduction, Prentice-Hall 1973.

(5) Teitelbaum, T., Reps, T.: The Cornell Program Synthesiser: a Syntax-Directed Programming Environment, Comm. ACM Vol.24, No.29,Sept.1981,pp563-573.

(6) Rose, G.A., Welsh, J.: Formatted Programming Languages, Software Practice and Experience, Vol. 17, No. 7, July 1981, pp651-669.

COMPUTER POWER:
A Coordinated Series of High School Curricula

Charles E. Hughes
University of Central Florida
and Gentleware Corporation

and

J. Michael Moshell
University of Tennessee
and Gentleware Corporation

ABSTRACT:

Most high school computer science courses have the unfortunate characteristic of associating computer skills with the elitist tradition of science and math curricula. This association scares away many students who would benefit from an ability to make effective use of computers. The Computer Power curricula series, a project funded in part by the National Science Foundation, uses a source-level interpreter for the Pascal programming language and a library of real-time graphics animation routines to introduce students to the fun of computing. In this environment the students use a modern programming language with excellent control structures to produce animation sequences. The application, animation, spans cultures, economic levels and academic orientations.

I. Introduction.

In 1978 several faculty members at the University of Tennessee teamed up with three local high school teachers to develop a novel approach to secondary school computer science education. This lead to a proposal and its subsequent funding by the National Science Foundation. The goals of this project were to create an introductory computer science course having the following characteristics:

...The course should be as accessible to the student who would normally avoid math/science courses as to the technically oriented one. It should not prey upon the fears of those many students (and teachers) who are afflicted by "math anxiety".

...Problem solving skills should be emphasized. While components of the course should address topics such as the societal impact of computers, this should not be the primary goal.

...The first programming language should be one that encourages good habits of top down design and information hiding. Pascal should be the primary language with BASIC being introduced only after a solid foundation has been developed.

...The course should take advantage of color graphics, animation and sound generation. Students should find the products of their work to be as compelling as the displays presented in video arcades.

II. Choosing Hardware and Software

When we selected the brand of microcomputer we carefully enumerated all those features that we felt were essential to the success of the project. Those deemed most critical were:

...Reliability -- We could not tolerate the situation where teachers and students become frustrated by machines that are down more than they are operational.

...Quick Repair -- We felt it unacceptable to have to send units back to the manufacturer and then wait three to six weeks before the repair is made.

...Color Graphics -- In order to be successful, the material presented must engage the student's attention. Black and white and/or unanimated graphics, while acceptable in many environments, are not competitive with colored animation.

...Pascal -- In addition to teaching Pascal to our students, we wished to use it for the development of all supporting software. This would increase the chances that the resulting products would be both reliable and portable.

At the time we first started evaluating computers (mid 1979) there was in our opinion only one system on the market having the above characteristics. This was the Apple II computer marketed by the Apple Computer Inc. The Apple had, and still has, decent resolution color graphics, sound generation and support for a dialect of the UCSD Pascal System.

After selecting the Apple we initially thought that the Apple Pascal System would serve as a good software environment in which our students could create, test and maintain their programs. We quickly learned that this was not so. While most satisfactory as a development environment, Apple Pascal has a number of features that are very bad for the beginner. In particular, simple typing errors when imputing numeric values can cause the system to "crash". It is also common to find yourself stuck in the editor with no way to save the current workspace. Finally, since Apple Pascal is a compiled language, it lacks the immediate response and friendly run-time error messages that are possible in interpreted languages such as BASIC. Nonetheless we found many admirable features in the Apple Pascal System. Thus, our first major software effort was directed towards developing a system that incorporated the good features of Apple Pascal, the interpretive environment of BASIC, and a wide variety of hand-holding and curricular support features. This system, which is discussed in the next section, was molded by curriculum needs, rather than the curriculum being designed to accomodate the esoteric features provided by some slightly mad computer scientist.

III. Developing Appropriate Software

As already mentioned we decided in advance of starting this project that Pascal would be the first language we would teach and that the Pascal system we acquired, or created if need be, would be oriented around the beginner and would support color graphics. After rejecting Apple Pascal due to the ease with which programs developed under it could "crash", we were left with the problem of deciding just exactly what were the precise features we needed. This turned out to be both a fortunate event, since we were able to create the new environment from scratch, and a curse since it meant many sleepless nights developing this new "utopia".

Before creating this new system, we tried to analyze what we liked and disliked about the Apple Pascal System, and why BASIC, despite its known faults, had achieved such a clear preemminence in the educational market.

Concerning BASIC, we found its chief advantage to have nothing to do with the language itself but rather with the environment in which one creates, tests and runs BASIC programs. The primary characteristics are the ease of program entry and execution and the availability of "immediate mode" commands (that is, you type one command and see the results immediately). We saw no reason why we could not provide such a facility in a Pascal based system. Furthermore, the creation of such an interpreted version of Pascal easily overcomes the"crash" problems of Apple Pascal since our interpreter is always in control, holding the naive user's hand.

Based on these observations we designed and implemented a system called INTERPAS. We provided immediate execution and set things up so that one can simply type a program and then say "Execute this". Moreover, we didn't stop there. We provided features that make the creation and manipulation of animated objects a simple task within Pascal programs. We provided program trace facilities, whereby a user may observe the program as it runs line by line. In this mode, the top 5/6 of the screen are used for graphics display and the bottom 1/6 trace commands. The association of cause (a Pascal statement's execution) and effect (perhaps a change to the graphics display) is evident.

The INTERPAS system has evolved a great deal in the last two and half years. Moreover, the animation support system, called Rascal (standing for Rascal--A Structured Computer Animation Language), has become a convenient, comprehensive "tool kit" for the development of animation sequences. It allows the recoloring, rescaling, rotating and repositioning of objects. It provides support for creating objects, painting background scenes on which to stage their performance, and scoring music. Rascal now serves as the basis for several new curricular activities that are discussed later in this paper.

The INTERPAS system follows the basic flavor and design philosophy of Apple Pascal and its parent system UCSD Pascal. The filer used to manage data on the floppy disk devices is very similar to that found in Apple Pascal. The editor is also similar but allows a simple transition to filer functions without losing the current edited work space. There is also a simple transition from editing to execution. A single keystroke does it. Completion of execution places you right back in the editor. A list of some of the more important features of INTERPAS follows.

...Understandable error messages -- These are always associated with the Pascal statement in error, whether the problem is a syntax or a semantic error. Single line correction of the offending statement is possible without reentering the full editing process.

...Executing as far as possible -- Under Apple Pascal any error in the syntax of a program will cause the program to fail to get into execution, even if the error would not be encountered until well into the program's execution. With INTERPAS, execution proceeds until the offending statement is encountered.

...Program tracing --When using INTERPAS, it is possible to trace both the progression through the program's statements and the course of values assigned to each variable.

...Aborting at any time -- One may press the control key and the "C" key, simultaneously, at any point during the execution of a program. This causes an immediate termination with control returning to the editor.

...Quick type for unskilled typists -- Most children and a surprising number of computer scientists, including one of the authors, are poor, one finger typists. INTERPAS provides a feature whereby each key on the computer's keyboard is associated with a Pascal or Rascal keyword. The pressing of control and Q simultaneously, followed by a desired single key is equivalent to typing an entire keyword. This provides a great speedup in program preparation time.

...Break out and back into a program -- A very special feature of INTERPAS is its ability to allow a user to cause execution of a program to be temporary redirected to be from the keyboard rather than the edited program. When you choose to resume execution, you are right back where you broke out of previously.

...Protection -- Both read and write protection are supported by INTERPAS so that students can protect their programs from the unauthorized perusal of others.

...Simpler syntax -- Although Pascal requires semicolons between contiguous statements, and INTERPAS accepts such usage, our system does not require them.

From the above it might first appear that INTERPAS has obviated the need for Apple Pascal. Not so. While convenient for learning Pascal, INTERPAS does not support the full range of features found in Pascal. Moreover, programs run much slower than when they are compiled. This is one of the reasons our system so closely matches theirs. This makes the transition easy when the student becomes ready to take on more extensive projects, such as in the second Computer Power course that we are now classroom teaching. For the case where a program has already been developed to run under INTERPAS there is a program that automates conversion to the standard Apple Pascal format.

IV. Computer Power I

Computer Power I is the flagship of the Computer Power series. It is a first computer science course for high school students. This course has been extensively tested for the past two and a half years. The original test sites were three schools in the Knoxville, Tennessee area. This year is serving as the evaluation year and as such there are ten test schools that are located in a mix of inner city, suburban and rural environments.

The common thread throughout all lessons in Computer Power I is that students learn by doing. Students get hands on experience from Day 1, and each is strongly encouraged to use the computer every day. Each lesson includes elements of exploration, skill building and design. The course represents one of the very few in which experimentation is not a cookbook experience with the results predetermined and passed down from class to class as are the results of typical classroom Physics "experiments".

Computer Power I is intended to be taught in a one semester (eighteen week) course. From the first week group interaction is stresses. In this first lesson students are divided into groups of three. While one group is using the computer, the others are analyzing the results of their last session and planning the work they will do during the next one. This first lesson does not involve our Pascal system but rather has them interacting with a playful system called Quilt. Quilt is a program that is easily understood by four year olds and yet is enjoyable for adults. There is practically no way to fail to produce striking, interesting patterns with the Quilt program. The interaction with Quilt is almost trivial:

...Keys around 'K' move and paint a dot;

...Paddle 0 changes the dot's color;

...A key causes the pattern to repeat.

With Quilt there is nearly always a beautiful pattern that proceeds from symmetry and repetition. As in other places in the course, we provide students with visually striking results in order to reinforce a sense of well-being and control. Still there are messages that subtlely teach many important concepts in problem solving, science, mathematics, composition and so on. For example, repetition of a sequence of moves and color changes does not result in the same thing being done over and over again. The world in which the actions occur changes based upon previous actions, and thus a copy of the pattern is started at the point where the last copy stopped. This mirrors the situation that occurs when a loop is executed within a computer program. Although the syntax of the loop's actions remains the same, the affect of carrying out these actions changes as a result of the cummulative affect of all previous executions through the loop. Beyond this there is the discovery process. We continually ask students to discover the parameters of the system, giving them as little verbal, a priori information as possible. We hope that evaluations will show that students are capable of transferring this ability and willingness to explore to other subjects.

After quilting, students are introduced to the Pascal language. Initially they use the "immediate" mode to type single commands, generally calls to procedures that provide animation, and observe the affects of these. After a short period of this activity, they are introduced to the editor and given simple complete Pascal programs to edit, execute, re-edit, execute, etc. The first programs they create involve no control structures but they are quickly introduced to the most important of all control structures, the procedure. Thus, modular design and top-down programming are taught at the outset, not after bad habits have already been developed. From this we progress through other control structures, REPEAT-UNTIL loops, IF-THEN statements, arrays and so on. For students who complete their work early there are eighteen sets of "Blue Sky Projects". These open-ended problems involve many fields of interest, including science and art-related topics. The "capstone" of the course is an individual final project and a "show and tell" evening in which students present these projects before an audience of parents, friends and local media representatives.

V. Other Computer Power Products

The first Computer Power course was such a satisfying experience that we are now developing other courses having the same philosophy. Computer Power II, a second semester course for high schools, is now in an early test phase. This course will develop highly skilled computer programmers who are well versed in Pascal, BASIC and the dynamics of large programming projects involving a team rather than an individual effort.

We are developing a self study version of Computer Power I. This involves the creation of a tutorial that leads its reader, in an interactive manner, through the lessons of a first course.

We have created a programming environment for very young children, generally four to eight years old. This will serve as the basis for an introductory course for elementary school children.

The animation capabilities of Rascal are now being used to deal with the special problems of learning disabled children. One experiment is using cartoons to provide an environment in which language disabled children may learn some of the powers of language.

In addition to full semester courses we are developing several mini courses. An example of these is a system that uses animation to allow students to experiment with and graphically see the results of modifying parameters that affect energy consumption in the home.

VI. Availability

Computer Power I is now being distributed by the Gregg Division of McGraw-Hill. For further information contact the publisher at 1221 Avenue of the Americas, New York, NY 10020.

All other Computer Power products are being developed by Gentleware Corporation, the firm that created INTERPAS and Rascal to support the first course. If you wish to be kept informed of the status of these projects write to us at Post Office Box 8301, University Station, Knoxville, TN 37996.

'Breaking a problem into smaller pieces will make it more manageable.' Well, if it works for a program, it should work with my project
VERY FUNNY!
Maybe it just needs to be broken into smaller pieces!

INTERPAS, Rascal and Gentleware are Trademarks of Gentleware Corporation. Registration has been applied for. Apple II is a registered Trademark of the Apple Computer Company

This project is partially supported by NSF grant SED-79-18991.

VII. A Sample INTERPAS Program

```
PROGRAM SPRINGTIME
(* A FLOWER BLOOMS.   *)
(* A BEE FLIES TO IT. *)

CONST BUMBLE = 1
      HEX = 2

PROCEDURE MAKEHEX
(* A HEXAGON IS USED FOR EACH PETAL *)
VAR   SIDES : INTEGER

BEGIN
 CREATE (HEX)
  TURNTO(0)
  POSITION(TURTLE,140,75)
  FOR SIDES:=1 TO 6 DO
   BEGIN
    MOVE(10)
    TURN(60)
   END
 ENDCREATE
END (* MAKEHEX *)

PROCEDURE MAKEFLOWER

VAR   PETALS : INTEGER

BEGIN
 (* DRAW STEM OF FLOWER *)
 POSITION(STICK,140,20)
 SETCOLOR(STICK,GREEN)
 SETANGLE(STICK,90)
 SETSCALE(STICK,75)
 PAINT(STICK)

 (* USE HEX PETALS TO DO FLOWER HEAD *)
 FOR PETALS:=1 TO 12 DO
  BEGIN
   PAINT(HEX)
   SPIN(HEX.30)
  END
END (* MAKEFLOWER *)

BEGIN
 MAKEHEX
 MAKEFLOWER
 GETFIGURE(BUMBLE,'BEE')
 SETSCALE(BUMBLE,75)
 POSITION(BUMBLE,240,75)
 TURNON(BUMBLE)
 REPEAT
  FLY(BUMBLE.-10,0)
 UNTIL WHEREX(BUMBLE)<=170
END
```

THE POLITICAL PLATFORM OF THE COMPUTER

Judith W. Leslie

Vice President for Planning & Development, Pima Community College

What can this candidate offer you, a member of the executive staff of an educational institution?

1. First, this candidate will save you money. How?
   - by lowering your personnel costs
   - by improving productivity of your faculty and staff
   - by improving your personal productivity.
2. Second, this candidate will give you what you need to do your job effectively. How?
   - by giving you the information you need to make decisions
   - by giving you information when you need it
   - by giving you alternatives to consider.
3. Third, this candidate promises to improve your expertise. How?
   - by freeing up your time from routine tasks
   - by teaching you new ways to do your work
   - by changing the way you think about your work.
4. Fourth, this candidate promises to improve communication within your institution and among institutions. How?
   - by establishing communication lines throughout your institution
   - by communicating promptly through these lines
   - by extending communication to other institutions and and agencies.
5. Fifth, this candidate promises to improve continually while in office. How?
   - by reducing costs each year
   - by working harder
   - by becoming more accessible with each succeeding year.

But what must be done for this candidate to be elected? Let's start with yourself; do you have any reasons why you would not elect this candidate?

Consider...

- Have you ever met the candidate?
- Have you read of the accomplishments of the candidate?
- Have you seen the candidate in action?

You have? And you have observed some problems:

- The candidate speaks in a language you don't understand well.
- You find the candidate is impersonal.
- It is difficult to get through the candidate's facade.
- The candidate's written communication is incomprehensible and it isn't written on standard letter size.
- The candidate's appearance is unusual, strange blinking eyes.
- The candidate has been known to make mistakes and even stop working.
- Only select people associate with the candidate.
- The candidate is primarily supported by the youth.
- You don't understand what makes the candidate tick?

But have you met anyone who has had a different reaction to the candidate? Your friend? And what did your friend have to say about the candidate:

- that the candidate was the most effective person he has ever known
- that the candidate helped him do his work more than anyone else ever has
- that with a little bit of study and practice the candidate's language is easy to comprehend
- that the more you are around the candidate, the more you appreciate the qualities of the candidate
- that your friend now could not do without the candidate's friendship.

That's quite an endorsement. Certainly worth your further consideration. Perhaps you could realize what this candidate can do for you to a greater extent if we discuss how the candidate can help you get a promotion.

I imagine that for you to do your job effectively you must have authority to carry out your responsibilities Let's examine on what basis your authority is bestowed. To do so, we shall refer to the work of Mortimer and McConnell, entitled Sharing Authority Effectively (1979). They refer to two types of authority. The first is "formal" authority, which is based mainly on hierarchical position and the sanctions inherent in a particular position; and second, "functional" authority, which is based upon factors such as expertise, competence, information, skill, legitimacy, and personal persuasiveness. One can infer from their research that in recent times, an individual in a leadership position is more likely to succeed if he/she exercises functional rather than formal authority. They state, "The president can no longer rely solely on formal authority, which is based mainly on hierarchical position and the sanctions inherent in a particular office...The president's task is to win consent, not to command it," (p168).

In discussing both formal and functional authority, it is helpful to discuss, in addition, the concept of power. In particular, what are the resources of power? Some of the more obvious ones are as follows: money, expertise, competence, position, information, human relations skills, credibility, time, and connections to others with these resources. Those resources most conducive to formal authority are position, money (and sanctions) and connections. The resources most conducive to functional authority in contrast, are expertise, competence, information, human relations skills, credibility, and time.

If one accepts the assumption stated by Mortimer and McConnell that it is considered more effective to gain compliance through the use of influence (functional authority) than by resorting to the authority inherent in a formal position, then the resources of power which support functional authority should be those you wish to acquire.

Let me tell you how this candidate can help you acquire these resources.

1. Expertise. The candidate has the capability to store, retrieve, analyze, produce and disseminate information that will add greatly to your expertise. The candidate also can access the expertise of others to assist you in building yours. In addition, by utilizing the candidate's capabilities, you also will acquire new skills and additional expertise...expertise that is not common as yet among administrators.
2. Competence. Through the information that the candidate can provide, you will

be able to make decisions based upon a comprehensive information base; you will be able to formulate alternatives and conduct simulations to evaluate the effect of each alternative. Further, your information will be organized systematically so that as you carry out your daily responsibilities, you will have immediate access to the information you need. As a consequence, the candidate will assure that your competence as an administrator is improved.

3. Information. You are living in an era of "information explosion." The organization, storage, and accessing of this information can be done by the candidate. You will have at your fingertips the information you need, perhaps even displayed in graphic form so that you can interpret it easily.
4. Human Relations. Although acquisition of this resource is largely up to you, the candidate can facilitate how and when you communicate. The candidate can link you to personnel throughout your institution as well as to personnel external to the institution. The candidate will assist you in responding promptly to those who do contact you. And, the candidate will perform a large portion of your routine tasks so that you can be free for personal contacts.
5. Credibility. The candidate can enhance your credibility through the precision, accuracy, and dependability that the candidate possesses. Further, your decisions will be rationally supportable by the information analysis made possible by the candidate.
6. Time. The candidate is very adept at saving you time. The candidate is able automatically, rather than manually, perform numerous tasks in which you are involved. The candidate can even manage your calendar.

In sum, the candidate can provide to you the critical resources of power to exercise effectively your authority within an institution. When you do, you may get that long awaited promotion.

Now, do you have any reservations regarding the candidate? You think you will vote for the candidate? But you still have a few remaining concerns, the ones you expressed initially from your own observations? Well, let's discuss them.

1. Language. The candidate is in the process of learning your language. A basic vocabulary has been acquired and soon the candidate will have mastered the sophisticated task of speaking your language. You must realize, however, how complex your language structure is, not to mention its various idioms and dialects.
2. Impersonal. The candidate is becoming much more friendly. The most notable personally warming attribute is the gamesmanship in which the candidate participates. People are getting much more familiar with the candidate now that these games are being played. Do you like football? Well, so does the candidate.
3. Facade. The candidate's image is changing dramatically. In fact, the candidate has lost a great deal of weight and has an entirely new wardrobe. With this new image, improved use of your language, and a little games playing, you

might more easily get through the facade.

4. Written Communication. Do I have a surprise for you. You know the saying, a picture is worth a thousand words? Well, the candidate has acquired quite unusual artistic expertise. The candidate can now transform that written communication into diagrams, charts, tables, figures, and pictures -- in color if you prefer. Yes, you'll like this new form of communication.
5. Blinking Eyes. With the candidate's new image, you probably won't even notice eyes at all. You'll be so interested in what the candidate has to say and portray, that blinking eyes may never even be noticed.
6. Mistakes, Work Stoppage. Just like you, the candidate has made mistakes and yes, the candidate has even been sick on occasion, to the point occasionally of not working. But just as advances in medicine are helping you to have fewer illnesses and improved treatment when you do become ill, the candidate's health is improving through advances in technology. Overall, the candidate is much stronger now than in the past, work performance has improved, and there are fewer sick days taken. The candidate may even outlive you.
7. Selective Association. It's true that in the past the candidate has had limited association with others. Realize of course, that this limitation was primarily due to the communication problem. Those who spoke the candidate's language have been very well received by the candidate. As the candidate learns your language and you learn the candidate's, your acquaintance will improve.
8. Youth Supported. You'll recall that I mentioned the candidate likes to play games. Well, you know who else does? The young people increasingly are spending their recreation time with the candidate to play these games. Recently, however, entire families have joined in and adults are finding that they too like to play games. In addition, once adults feel comfortable with the candidate, they'll find the candidate can help balance the checking account, do personal shopping and the candidate can provide the daily stock report to you. I can see that you're interested in those characteristics.

What makes the candidate tick? Perhaps I could ask you if you know what makes your spouse, friend, or child tick? Complex, isn't it? And yet, the candidate is less complex than they are. Just as with people, though, the more you are around them, the better you get to know them. So it is with the candidate.

Now may we have your pledge of support for the candidate? For if elected, this candidate will not only help you, your institution, and the field of education to move ahead, but also will revitalize America as a whole. Her economy will improve, her work force will again make her a world leader. Join with us ... elect the Computer!

A SUPERINTENDENT'S PERSPECTIVE ON MANAGEMENT INFORMATION SYSTEMS

In 1973, after a two-year study, the Florida Governor's Commission on Education issued a report containing the following statement of objective and intent:

> "Move the process for decision making as close to the student as possible."

The statement was written into law that year by the State Legislature.

The 1976 Legislature expanded on that concept by writing into the Educational Accountability Act of 1976 a requirement that:

> "Each school district board shall utilize its system of planning and budgeting to emphasize a system of school-based management in which individual school centers become the principal planning units and eventually to integrate planning and budgeting at the school level."

The Legislature also wrote in extensive processes for parental and community involvement.

The Legislature and the State Board of Education implemented this management concept by repealing a great number of regulatory laws and rules and giving local school boards more flexibility and authority in the decision-making process. It was contemplated that this greater flexibility and authority would be passed through the district to individual schools.

Implementation of the school-based management concept will significantly alter the way in which school systems have been traditionally managed. Correspondingly, it will alter the way that school-based and district-based personnel relate to each other and substantially change their functions.

Therefore, when we consider the specifications of a management information system, we must view those specifications within the framework of how a particular district is organized and managed. An information system designed to support management in a decentralized system will be different from a system designed to support management in a centralized system. It will be considerably more sophisticated and complex.

There is no question that the State was determined to create a decentralized system of school-based planning, budgeting, and management and to inject citizens into that process in a meaningful way. But an interesting question is why the State moved in that direction. Why were laws passed that mandated a fundamental shift in the way that school systems have traditionally been managed?

There appear to be four basic reasons for this action.

1. School-based management with significant citizen participation is a prerequisite to developing an adequate response to the diversity of student needs and a provision of equal educational opportunity.

Students are unique. They differ in size, [illegible], maturation levels, handicaps, motivation, aspiration and learning style. The list of differences is endless. These differences say that we do not have standardized students. Unique students add up to unique classrooms of students; unique classrooms of students add up to unique schools. The irony of this is that it has to be said. It has to be said because while we have recognized individual differences in students, classrooms and schools, we have not adapted school management to this uniqueness, nor have we developed a management information system to support it.

If we accept the premise that students, classrooms, and schools are unique, then we must consider where best to respond to that uniqueness. Do we respond to it best at the district level or at the school level?

Central office personnel are simply not in a position to perceive and accurately respond to the diversity of student needs throughout a school district. Such a response is more likely to succeed where the staff is in personal communication with students and where needs of the particular students are known.

That being the case, it would seem to follow that school-based personnel should have the authority and flexibility to respond to diagnosed needs of students with prescriptive programs. There is no higher order of professional response than for a principal with the managerial skills to involve his or her staff, parents, and community in a process of analyzing the unique needs of a student body; designing a program to meet these needs; allocating resources to employ and deploy staff, assign space, procure materials, supplies, and equipment to implement that program; and then evaluating how well they did.

The management information needed for a principal to perform that highly complex task is considerably different from that of a principal receiving an allocation of personnel, materials, supplies, and equipment based on a district allocation formula.

Equal educational opportunity in the constitutional sense says that the quality of a student's education must not be dependent on the local wealth or lack of wealth of school districts. Most legal authorities believe that the Florida funding formula which funds individual students according to need and nullifies local wealth, can withstand a legal test of equal educational opportunity. The corollary is that if the funding formula can stand the test of equity interdistrict, it can stand the test of equity intradistrict if the district flows through to schools a pro rata share of the base student dollar receipt from the finance program in a manner that tracks the State funding formula. This in effect is school-based budgeting, in which a school's budget is determined by the unique mix of a student body and funds flow to match that mix.

It seems clear that management information at the school level where school-based budgeting has been initiated must be a great deal different from what would be required in the case of a district that has not instituted school-based budgeting.

It has been said that a simple legal concept of equal educational opportunity and an equitable flow of funding within a district is not sufficient — that school boards must do more to insure that students from school to school are receiving educational equity. Most would agree. The point of disagreement comes with regard to the method of achieving equity. This consideration leads to the second reason for establishing school-based management.

2. School-based management with significant citizen participation is a prerequisite to school-based accountability for the educational product.

There is general agreement that there should be school-based accountability. The question is — Accountable for what? A simple answer would be that school-based personnel are accountable for implementing a district's program of education.

The problem with that is that a program can be stated in two forms: process standards and product standards. Process standards are "means." Product standards are "ends."

Process standards tell managers "how-to-do-it." They tell principals how they must organize their school; how many teachers, aides, library books are allocated to them; how many minutes and periods per day there should be; and so on. The output or product of this process is not addressed. It is assumed that if the recipe is followed a satisfactory student performance will result. Process standards say, in essence, that we may know how to get there but we do not necessarily know where we are going.

Product standards, on the other hand, state the ends, the output, the quantifiable level of student performance that the process is to produce. Product standards say that we know where we want to go and we hold managers accountable for getting there in the most cost-effective way.

Program stated in terms of process standards is a centralized mode of operation. Program stated in terms of product standards is a decentralized mode of operation. Process standards regulate the method of educating which determines the product, whatever it might be. Therefore, those who regulate the process are accountable for the product so long as the manager correctly follows the prescribed method. If a board prescribes a process and principals correctly follow that process, then the board, not the principals, is accountable for whatever that process produces. Those who regulate the process are accountable for the product.

On the other hand, if the board states the product in terms of a progression of competencies that students in a system are to acquire and leaves the management of the process to principals, then principals and not the board are accountable for the product.

The management information system required to permit teachers and principals to track students through a progression of competencies is considerably more sophisticated than one that produces status reports on process standards. Similarly, the information system required to enable district managers to monitor the progression of students for the purpose of instituting a system of educational accountability is a different system from that required to monitor process standards.

3. School-based management with significant citizen participation is a prerequisite to constructive change and improvement in education.

The third premise is that school-based management is a prerequisite to achieving constructive change and improvement in education. That is based on the assumption that constructive change and improvement will not be achieved without broad-based citizen understanding of education, its strengths, its weaknesses, and its needs. If that is the case, how do we get that broad-based understanding? We get it by meaningfully involving parents and community in the educational process. How do we get meaningful participation in the educational process? We get it by involving parents and community in their local school — where participation is relevant. How do we get the community meaningfully involved in its local school? Certainly we will not get parents and other citizens meaningfully involved at their local school if the only decision in which they participate is whether the school needs a movie projector more than an overhead projector or whether they are going to sell cookies or raffle a cake to purchase [illegible]

Meaningful involvement happens when people participate in making important decisions. From participation in making important decisions comes understanding. Understanding generates support and commitment, which in turn can bring about change. But how can a principal involve parents and community in making important decisions if he or she has no important decisions to make?

4. School-based management with significant citizen participation is a prerequisite to retaining local control of education.

The fourth premise is that school-based management with significant citizen participation is a prerequisite to retaining local control of education.

Two powerful forces are at work that threaten local control of education. First, the funding source for education is moving from the local community to higher levels of government. Secondly, vested interest groups are gaining sufficient political power to control the election process.

With regard to funding, local communities vary in wealth and therefore in ability to support quality education. The concept of equal educational opportunity cannot tolerate that variance. Thus, states are taking over the major funding responsibility for education. The questions are whether control of education will track funding to its source and, if so, how local control of education can be retained when the source of funding is at a higher level of government.

In the political arena, the only force with sufficient power to counter the growing power of vested interest groups is a mobilized public. How does one mobilize the public? One way is to involve the public in a manner that it gains understanding of the issues. That gets back to community involvement at the school level where the community participates in the educational process in a manner that provides insight, understanding and commitment. That insight, understanding and commitment will flow to boards and legislators as a political base from which they can act independently of parochial pressures.

Therefore, it does not seem far-fetched to suggest that school-based management with significant citizen participation is an essential for

--meeting the individualized needs of students

--implementing a program of school-based accountability

--bringing about major improvements in education

--maintaining local control of education.

However, if school-based management is to succeed, school systems must develop an information system that is much more sophisticated than most districts currently perceive as being necessary for effective management.

TITLE: A Superintendent's Perspective on Management Information Systems

DESCRIPTION: The presentation develops four basic reasons for the State's legislation creating school-based planning, budgeting and management with citizen endorsement.

1. It meets more effectively student needs and provides equal educational opportunity.

2. It brings about school-based accountability for the educational product.

3. It brings about constructive change and improvement in education.

4. It retains local control of education.

The presentation concludes that a more sophisticated information system is needed if school-based management is to succeed.

X Administrators - Elementary, Middle and Secondary

- William McFatter

MICROCOMPUTERS: GETTING STARTED IN A JUNIOR HIGH SCHOOL

Mary M. Pinski

Herrick Junior High School, Downers Grove, Illinois

Introduction

Computer programs are springing up in schools all over the country. These programs, depending on funding, expertise, and objectives, are varied in their organization and curriculum. The computer literacy program in School District 58, in Downers Grove, Illinois, was initiated in September, 1981, after a year-long study.

The Committee

Are microcomputers needed in the elementary schools? This is a question that has plagued many school districts in recent years. District 58 set up a microcomputer committee to study the use of microcomputers in education. The committee was comprised of administrators, teachers, parents, and Downers Grove residents who all were, in one way or another, interested in microcomputers. The main objectives of the committee were to decide whether or not microcomputers were necessary and to make recommendations to the superintendent.

To meet the first objective, the committee divided into various teams. These groups investigated various computer programs in the surrounding districts. Visitations were scheduled and an informational fact sheet on each program was completed. Information such as type and number of hardware, number of students, specific program, benefits, and problems encountered was brought back to the committee as a whole and evaluated. Each team also had another responsibility. One group concentrated on why computers were important and did research to support that idea. Another group investigated the use of the district's only microcomputer, a Texas Instrument purchased by the PTA of one of the elementary schools. The last team wrote a proposal for the use of microcomputers in the junior high. This included the investigation of hardware.

It was agreed upon that our report should show a cautious approach. We did not want to rush into anything and have the microcomputers sitting in a closet collecting dust. We, also, did not want to force them on unwilling teachers.

The Report

Our final report had three parts. The first was a statement as to why the committee believed that computers should be introduced in District 58. It said, in part:"Essentially, the committee feels that from both a philsophical and practical standpoint, it is essential that our students have exposure to computers prior to high school because of the all-pervasive influence of computers on their future."

The second part of the report contained our proposal. Our primary focus was to be on computer literacy for all 7th and 8th grade students in the two junior high schools. We included recommentations for purchase of hardware and evaluation procedures. The proposed budget was $10,000.

Finally, a statement on the use of microcomputers in the elementary grades was given. It was felt that at this time, it was not feasible to introduce microcomputers in the elementary schools. It was suggested that the one school currently using a microcomputer report on its use in the upcoming year.

We decided to recommend purchasing the best system we could with the money budgeted. Each junior high would have the same equipment. A Bell and Howell Educational Model 48K Apple was our hardware choice. The safety feature of the locked lid and the convenience of the power pack on the back made this the choice over the regular Apple. Two black and white monitors were selected for each school. Since they were less expensive than color, this enabled us to purchase two. These monitors could then be placed in different parts of the classroom so that everyone could see. A disk drive system and a printer completed our hardware request. The remainder of the budget was to be used for a summer workshop for the eight junior high teachers and for purchase of software.

The proposal was submitted by the superintendent to the school board. Approval was given at a June meeting.

Program Preparation

During the two week summer workshop, the math teachers developed objectives and set up a program. Although all eight teachers had some experience with computers, most were unfamiliar with microcomputers, and a great deal of time was spent in becoming acquainted with the equipment.

In setting up objectives, we examined the "Computer Literacy Objectives" that were set up by the MECC Computer Literacy Study. These included: identifying parts of the computer system, learning about the history of computing, understanding what a computer can and cannot do, and to "not feel fear, anxiety, or intimidation from computer experiences."

We agreed that it would not be beneficial to us to write our own program for this first year because of being unsure of exactly where we wanted to go. After examining various texts available, we chose My Friend - The Computer by Jean Rice. It met our overall objective of computer literacy. It was easy to read and understand. The teacher's guide came complete with activity sheets, tests, and an evaluation, which we modified to suit our needs.

The Program

After just one year in operation, our computer literacy program has met with success. Although the junior highs differ somewhat, the program was basically the same for each. Each of our students had the course which ran for approximately 2½ weeks during their math class. During this time, each student had some hands-on time. Although it was minimal, it was enough for them to practice loading a disk, write a short program, and get a computer printout. This phase of the program took place during the first semester.

Before the beginning of the second semester, more hardware was acquired. The PTO had become very enthusiastic about our program and purchased another computer and disk drive. The school purchased two color monitors.

The second semester was spent with the computers rotating between math classes. Students were able to work on various software disks, both teacher created and commercial. Many students wanted to learn how to program and the need for self-teaching material was evident. Through investigations, we purchased Creative Programming For Young Minds, an excellent self-teaching guide to learning programming.

A couple of mini-courses were also tried out during this period. One of the math instructors taught a group of pre-Algebra 7th graders beginning programming. This was a one week class that was arranged with the switching of students with another math teacher. A resident of Downers Grove came in once a week and continued with programming with the pre-Algebra students. Lastly, a retired teacher, working for the district on an early retirement contract, came in once a week and taught beginning programming to our advanced and interested 8th graders.

Involvement didn't end with the students. A necessary item was to get the faculty and staff interested. Two early dismissal days were used as inservice to acquaint them with the equipment and materials available. They were encouraged to take classes after school offered by the high school and to take the microcomputer home over the weekend.

The microcomputers also became an important part in evening activities.

During Open House and Family Night, the computer was shown to parents and they had the opportunity to get some hands-on time. A school board presentation was made in the winter to help inform the public as to what was happening.

The Future

Plans for our second year are being formulated. A proposal has been made for the purchase of three microcomputers for use in the ten elementary schools. They would be used with the 6th graders for computer literacy. This will necessitate a change in the junior high program.

Many questions still remain that need to be answered: What kind of program should be offered to the 8th graders next year? Should the time continue to be taken from math for the computer or should it be a separate class? What additional staff training is necessary? Before the start of the next school year, these questions will be examined. Once these are answered, more questions will arise and the trend will continue.

We've past the first hurdle. We've got our foot in the door. We don't know, yet, where any of this will lead but we do know that a bright future lies ahead.

References

Brown, Devin. Creative Programming for Young Minds. Charleston, Illinois: Creative Programming, Inc., 1981.

Johnson, David; Anderson, Ronald; Hansen, Thomas; and Klassen, Daniel. "Computer Literacy-What Is It?" Mathematics Teacher, February, 1980, pp. 91-96.

Members of the District 58 Microcomputer Study Committee. "Microcomputers in District 58," A report, Downers Grove, Illinois, 1981.

Rice, Jean. My Friend - The Computer. Minneapolis: T.S. Denison & Company, Inc., 1981.

LONG RANGE PLANNING FOR COMPUTER USE - A TASK FORCE MODEL

S. M. Raucher and T. J. Koehler
Department of Management Information and Computer Services
Montgomery County Public Schools
Rockville, Maryland 20850

Abstract

In an effort to deal with the challenge of computer technology, Montgomery County (Md.) Public Schools (MCPS) commissioned a Management Operations Review and Evaluation (MORE) study of the Department of Management Information and Computer Services. This study was completed in the fall of 1980. Its results were presented at the 1981 AEDS Convention and were also summarized in an article in the Summer 1981 issue of the AEDS Monitor entitled "Management Operations Review and Evaluation of a School District Computer Services Department."

The MORE Study strongly recommended that MCPS develop a long-range plan to meet the computer needs of schools and central offices. In response to this recommendation, the Superintendent appointed a Task Force for Long-Range Planning for Future Use of Computer Technology. This paper outlines the methods used by the task force and the results and recommendations of the study. It describes the goals of the task force, the activity phases necessary to meet these goals, the technological environment within the county educational system and the functional requirements of the educational system. As a result, the task force activities provided a direction and a plan for both administrative and instructional computer use.

The authors believe that the task force methodology, representatative composition and "advocacy" structure provides a model for similar planning activities in other educational institutions. It is in that spirit that the paper has been developed.

OVERVIEW

Forecasters of the educational future predict radical changes in the way public education performs its assigned task. In a presentation to this year's National Computer Conference, a U.S. Department of Labor official predicted that by 1985, 80% of all jobs will require the knowledge of how to use or interface with computers. Already, computers can be found in over a million homes, and experts predict that by 1990 this figure will be over 20 million. Yet public education is lagging sadly behind these developments in society. Not only must public education prepare its students for a world of computers, but educational administrators are in desperate need of the added information power that computers can offer. The public's demand for accountability and government's increasing demand for reports are burdening an already busy educational staff. In an era of fiscal restraint, administrators must have the best possible data if they are to manage their declining resources effectively.

In an effort to deal with the challenge of computer technology, Montgomery County Public Schools (MCPS) commissioned a Management Operations Review and Evaluation (MORE) study of the Department of Management Information and Computer Services. This study was completed in the fall of 1980.[1] It strongly recommended that MCPS develop a long-range plan to meet the computer needs of schools and central offices. In response to this recommendation, the Superintendent appointed a Task Force for Long-Range Planning for Future Use of Computer Technology. This paper describes the method and results of this process.

TASK FORCE ACTIVITIES

Goals

The MCPS Task Force for Long-Range Planning for Future Use of Computer Technology was asked to develop a plan that could be used as a basis for decision-making for the next seven years, beginning with the FY 1983 budget. The specific goals were:

1. To assess school system management information requirements.
2. To evaluate all current and planned information systems.
3. To develop a strategic plan for the future use of technology.
4. To apply resource constraints by providing multiple options and the timetables and costs of each option.
5. To report to the Superintendent with a documented long-range plan and a short-range tactical plan (1-2 years) in time for making decisions for the FY 1983 budget (i.e., August 1981).

Organization

In order to represent fully all areas of the school system and to make the best use of personnel resources, the task force was organized as three functional subcommittees coordinated by a steering committee.

The Task Force Steering Committee consisted of the school system's top operating officials -- three Associate Superintendents responsible for Supportive Services, Instruction, and Special Education, the Directors of Accountability and MIS, and each functional subcommittee chairperson. These three subcommittees represented the following areas:

1. Sub-Committee on Instructional Use of Computers
   The group's scope covered direct use and study of computers by students and teachers in the classroom and the use of computers to support teachers and administrators in managing instruction. The Instructional Subcommittee consisted of the Director of Computer Related Instruction, (chair), Instructional Specialists, special education staff, teachers, and school-based administrators.

2. Sub-Committee on Pupil Services
   The group's scope covered recordkeeping and any activities related to student and school data necessary to operate the school or school system. The

---

[1] Raucher, Steve and Koehler, Tommas; "Management Operations Review and Evalution of a School District Computer Services Department," AEDS MONITOR, July/August/September 1980, pp. 8-10, 22.

Pupil Services subcommittee consisted of a Supervisor of Secondary Instruction (chair) elementary and secondary principals, special education supervisors, and the coordinators of state reporting.

3. Sub-Committee on Business Services - The group's scope covered school system administrative tasks not directly related to student data, e.g., activities relating to personnel, finance, budgeting, and facilities planning. The Business Services subcommittee consisted of the Director of Financial Services (chair), the Directors of Salary Administration and School Services, along with other business management specialists.

The subcommittees also included a cross fertilization concept where a business specialist also sat on the Instructional Subcommittee and vice versa.

Involvement of MIS Personnel

One of the key concepts of the Task Force was the "low profile" of MIS and technical personnel. The Task Force was specifically intended as a user group. Technical discussions were avoided, and all of the technical issues were handled as staff work for the Task Force. With the exception of the MIS director, no data processing staff served on the Task Force or its subcommittees. However, senior MIS staff were detailed as technical consultants to each subcommittee, and the entire MIS staff were drawn upon for staff work.

Activity Phases

The Task Force identified 30 specific activities needed to reach its goal. These events were grouped into three phases.

Phase I
Identification of Needs

The function of Phase I was the organization and training of subcommittee members and "user advocates." Inservice training sessions were held for staff to learn how to organize their particular data processing requests into a semi-standard format. Staff were also taught definitions of computer technology, and given the opportunity to expand their knowledge of the potential uses of technology.

During preliminary meetings, each subcommittee read and heard testimony from the "user advocates". A total of 90 user advocates submitted 255 separate need requests to the appropriate subcommittees. Subcommittees organized requests into clusters of needs upon which the application of technology would be based. Each subcommittee then defined and documented benefit measures for its functional area. Subcommittees set priorities for each application, based upon this criteria of importance to the user, i.e., benefit without regard to cost. Phase I was concluded by drafting descriptions of general and functional environments and making an outline of the final report. Each of these efforts was evaluated during a Superintendent's Review at the end of Phase I.

Phase II
Prioritization of Needs

In Phase II each subcommittee analyzed application requests from the other groups and, if necessary, set new priorities for applications. This was necessary because the functional concerns of some applications overlapped (e.g., requests for tuition applications involved both Pupil Services and Business Services). The Computer Services Department then estimated the resources needed to implement each appplication. The subcommittees then developed multiple levels of requests according to economic constraints, i.e, minimal, medial, or ideal economic conditions. They then assessed the impact and consequences of implementing a system at each level. This phase concluded with revisions of the drafts of the subcommittee reports to the Superintendent.

Phase III
Final Report[2]

In Phase III the work performed by the Steering Committee paralleled that of the subcommittees in Phase II. The Steering Committee reviewed the subcommittee recommendations, defined the management plan (division of responsibilities, review and approval process, MIS monitoring procedure) and applied resource constraints by recasting the subcommittee recommendations into a Recommended Plan and a Basic Plan. Subcommittees reviewed and amended these plans and reviewed the draft reports. The Steering Committee then prepared the final recommendations and plans and approved the draft reports.

During each phase, the Task Force received continued support by executive management, as demonstrated by the Superintendent's and Associate Superintendents' attendance at task force meetings and monthly progress reviews.

The MCPS Technological Environment

The Task Force report found that public education, including MCPS, is in a period of self-evaluation. The MORE studies and the efforts of this task force were cited as concrete examples of this self-evaluation. As a result of this type of introspection, public education will doubtless conclude that it has indeed lacked understanding of the efficient use of available technology; that proper use of such technology can serve to tone up educational and administrative muscles; that computer technology has the capability of using resources more efficiently, providing significant assistance to the physically handicapped, providing information for better management decisions and public accountability, while at the same time contributing a necessary ingredient to the educational program.

A careful examination of recent trends, both internal and external, and trends in data processing led the Task Force to just such a conclusion. MCPS had been considered a national leader in the use of computers in the 1960's and early 1970's. Instructional uses and administrative systems were being developed and promised significant benefit to the school system. Unfortunately, the budgetary constraints, that began in

---

[2] Montgomery County Public Schools, Report of the Task Force On Long Range Planning for Future Use Of Computer Technology, August 1981. MCPS.

FY 1976, reduced the human resources available to implement such systems and removed the flexibility to use better performing equipment. The Task Force found that the school system today was utilizing the same systems and equipment available in 1976. It was also paying the price of high maintenance costs and inflexibility of its computer systems. While significant effort has been expended to maximize the benefit available from these "old" systems, the need to replace them was clear.

Trends in Data Processing

In 1956 only a few very large research institutions owned a computer. Today, elementary school children sit in their homes in front of a keyboard connected to more computing power than that of some of the larger computers of 1956. Today computers are smaller in physical size, larger and faster in computing power, and cheaper than computers of the past. This trend is expected to continue or even accelerate.

The Task Force found that in the face of this tremendous advance and great potential, many educational institutions, including MCPS, lagged behind. While some progress was made in establishing a pupil database, the majority of MCPS' current systems were far from being technically modern. Simple acquisition of computers with more power would not correct this lag. The Task Force identified several factors which tend to inhibit greater use of computers unless directly confronted. First, there was a widespread need for increased knowledge of computer uses and operations among the MCPS administration and staff. Before staff can fully communicate their needs, they need to appreciate the level of commitment necessary to attain the desired results. Lack of this "computer literacy" led to a skepticism of the value of data processing within MCPS. Contributing to this skepticism is the history of inaction, false starts, and failing systems brought about largely by this failure to adequately commmunicate the requirements of the application on the one hand and the lack of adequate budgetary resources on the other hand. The third problem to be overcome was the cost of maintaining the current system while a replacement system is being developed. On the other hand, if a decision is made to modify the current system rather than start anew, the problem will be further complicated for future generations. As such, the Task Force recommended that MCPS must view expenditures for increased technologic support as a long-range investment rather than as a short-range cost.

Functional Requirements for Technology

The Task Force recommended that MCPS expand technological support for both the instructional and administrative aspects of the school system. The specific requirements of these two areas vary greatly and were considered separately.

The instructional system must provide classroom experiences that increase computer literacy, classroom support systems, curriculum management support, better management of instructional resources, and a cost-effective method for collecting and assessing systemwide data on student performance.

The administrative system (including those for pupil services and business services) must provide a unified MCPS management information system with interdependent applications. Systems such as report cards, scheduling, transcripts and class size must use the same database. Old systems must be revised and these "foundation" databases must be established/expanded.

As a result of this duality, the task force concluded that separate networks need to be developed for the instructional and administrative systems. The instructional uses of technology can best be served by the use of microcomputers in each school. The plan thus calls for installation of microcomputers in each classroom, in sufficient quantity that each elementary school student will have 50 minutes per week of hands on experience, each junior high student 90 minutes per week and each senior high student 135 minutes. Translated into machines, this would require 3202 microcomputers in MCPS classrooms.

In addition, enhanced configuration micros, with bi-directional communications capability, special devices for the handicapped etc., would be placed in each school media center. This would bring the total number of micros to 3,862 by 1990.

The recommended administrative computer configuration is a distributed network with a strong central mode. Conceptually, data access should be available from any terminal regardless of the physical location of the database. As such, each senior high school would be equipped with a mini-computer that would contain skeletal data base records for its students, and those students currently enrolled in its feeder schools. Terminals would be installed in each school and communicate through the mini-computer. Those data requests/applications that can be serviced from the mini-computer will be accomplished without central lost interaction. Where service cannot be done at the mini-computer, data would "pass through" to the central mode. The central mode would be responsible for periodic (not less than daily) back-up of all distributed data, systemwide data synthesis and reporting, and operation of applications deemed inefficient for distributed processing, e.g., scheduling simulations.

The implementation of a management information system is an evolutionary process that will take years to complete. Initial activities must set the overall parameters, determine a flexible yet comprehensive systems architecture, and provide a foundation that will eventually support the entire system. Thus, priority must be given to those activities that will establish the foundation of the new system. The cost of these activities must be considered in light of their long-range value and not measured by current specific applications. For example, the cost of entering student addresses must be judged in terms of its long-range information system value and not measured against the cost of generating mailing labels. Priorities for applications which do not necessarily add to the foundation database should be set based on the application's benefit to the widest number of users and functions.

Benefit Criteria

The Task Force recognized that the standards for assessing the benefit of technological support vary dramatically between functional areas. While some business applications may clearly demonstrate cost savings, pupil service applications may derive more indirect benefit (such as improved efficiency of school-based operations). These cost or quasi-cost benefits cannot be easily compared with improved instructional quality or expanded student computer literacy. Because of the internal competition for MCPS resources, the task force recommended:

1. Functional benefit assessment criteria which are unique to instruction, pupil services, and business services functions must be adopted.

2. The executive management of the school system must be continually involved in reviewing the priority of applications and in allocating resources among the functional areas.

Results

The Task Force Report was submitted to the Superintendent of Schools in August 1981, and briefings were held for executive staff. The Superintendent provided the report to the Board of Education, and a presentation of the findings was made to the Board.

The Board subsequently requested detailed briefings on current activities and development methods, which were to be provided in two parts; one briefing on administrative uses of computer and one briefing on instructional uses of computers.

The report stimulated staff and community interest, and other presentations were held for the PTA and staff. Press interest resulted in several news stories highlighting various recommendations in the report.

Shortly after the receipt of the report, the Superintendent directed that the management recommendation of the report be implemented, and that the report serve as the basis for the FY 1983 budget.

CONCLUSION

While the degree to which the Task Force's recommendations will be funded is unclear at this writing, it is an accepted fact that the planning methodology and results are of major benefit to the school system. This task force represented the first planning activity to do long range planning outside of the annual budget development process. It brought together representatives from all segments of the organization, and gave executive level management an opportunity to concentrate on planning and technology. As such, the task force provided the forum for assessing current and future requirements for computer technology forced diverse operational units to produce a cohesive strategic direction/plan and provided the "springboard" for management review and decision-making for the near and long term.

USING MICROCOMPUTERS TO IMPROVE YOUR HEARING IMPAIRED PROGRAM

Susan L. Sarapata
Township High School District 214
Mt. Prospect, IL 60056

Abstract

The instructional and administrative applications of microcomputers in a high school hearing impaired program will be addressed in this presentation.

The approach described involves all members of the educational process of a hearing impaired high school program - hearing impaired and mainstream students, their teachers, and administrative personnel. The system has generated a high level of interest, involvement and productivity, and is able to accommodate the diverse needs of the educational community.

Background

The Hearing Impaired Program of Township High School District 214 is housed in one of eight district high schools. The program currently services 80 hearing impaired students with various learning abilities. They include those who have multiple learning problems and who are entirely self-contained as well as those who are completely mainstreamed and may require only the use of an interpreter. The staff of the program now consists of one supervisor, two counselors, two speech therapists, 15 teachers and five interpreters. The various classrooms of the program are located at numerous points throughout the building. This composition is an ideal situation for establishing a broad range of computer services because of a large number of needs to be met.

In the Spring of 1979, it was proposed that the Hearing Impaired Program would acquire three Apple II Plus microcomputers with disk drives and monitors. This equipment was to be used for student, teacher and administrative purposes.

The equipment was received in the Fall of 1980 and the remainder of that school year was set aside primarily for program development. This included the authoring of student courseware, developing a course outline to be implemented in 1981 and fostering general literacy of teachers and students.

The following school year (1981-82) was intended for implementation of the plans made the year before and continued planning for the future.

The Hearing Impaired Program came to microcomputers with little or no expertise in the computer field. I was selected to develop this aspect of the program because, as a college student, I had had experience using a small computer while working with graduate students. I had taught myself a computer language through a programmed learning manual and, using problems I designed, was responsible for running studies for students. This experience was not extensive and it may be interpreted to mean that the Hearing Impaired Program was starting at a deficit. However, the lack of structure which was brought to the situation has allowed the program to be flexible and evolve into a number of different areas.

How the Program Currently Functions

Resource Room

The computer resource room is set up to accommodate teachers, students and classes which need to use the computers. Last fall, I met individually with each teacher in the program to review software which I thought would be appropriate for his or her classes. It was then the responsibility of the teacher to set up a time to bring in students. Currently, the areas in which the computers are being primarily used are math, English and life skills. Some teachers have set up individual study programs with specific students. Now that some of the teachers are comfortable with the computers, some of them are reviewing new software that comes into the program.

Students who are taking classes in computing technology also have access to the resource room. The Hearing Impaired Program offers a self-contained course in Introduction to Computing Technology. This is a literacy course which covers history, current use, future trends, social issues, careers, word processing, logic development and programming in BASIC. This class is taught at a number of different levels to meet various learning needs. In addition, the class is taught in cooperation with a mainstream computer teacher in a mainstream setting. In this way, the program accommodates the wide variety of situations encountered in educating hearing impaired students. Further, students who demonstrate the desire and the ability to continue can register for more advanced computer programming courses in the mainstream setting.

Software Design

The original proposal to acquire microcomputers for the Hearing Impaired Program included use of the Apples for administrative purposes. During the 1980-81 school year, the program was focused upon the students. Math software was designed for and used by the students. This school year, the focus has been upon the design of software for administrative and general program use.

Again in cooperation with the mainstream computer teacher, higher level programming students are working with the Hearing Impaired Program to write the programs which are designed to be used by the Hearing Impaired staff. This approach has the advantage of giving students the opportunity to structure their own time while working under flexible supervision. The software currently being written is a Student Information File which records, updates and prints out information about a student's Individualized Educational Plan (IEP) and general testing status. The program also lists students currently in need of testing in various areas such as psycho-educational or audiological evaluation.

Plans for the Future

Ideally, the computer services of the Hearing Impaired Program will continue to grow and expand in the areas that are now being established. Given that administrative and other support remains consistent, the following expansion is projected.

Resource Room

The resource room will be made available to students, teachers and administration in a number of different ways. Use of the resource room will expand as more courseware is acquired and as more students enroll in computer courses. Teachers will continue to use the computers with classes or for individual students. Teachers will also have access to the room for software preparation or review and, if interest is generated, for creating programs using author languages or programming. Programming students will continue to have access to the resource room during study halls or before and after school. The resource room will also continue to be used as a teaching room for computer and math classes.

As an extension of the resource room, additional higher level programming classes, team taught by a teacher of the hearing impaired and

and a mainstream teacher, may be included.

Information Service

Present trends in designing software for the Hearing Impaired Program indicate that the computer resource room will also serve as an information service for teachers and administration. The previously mentioned Student Information File is one example of software to be used by the program supervisor. Additional software is presently being designed. A method that will cross reference and inventory teaching materials is needed. The Hearing Impaired Program is large and materials are scattered; many materials can be used in multiple teaching areas. Another need is for a current, centralized file to be set up for special students who require constant monitoring, frequent parent contact and the input of a number of different teachers and personnel.

An offshoot of the Information Service would be an in-school job for a student who could work with data entry and word processing. This student could go through the introductory computer course as a prerequisite and could learn other aspects of the position on the job.

Components of Effective Implementation

The successful use of microcomputers in the Hearing Impaired Program of High School District 214 has come about for several different reasons. Many efforts have come together to produce substantial results.

The district itself has made the commitment to computer education. When the Hearing Impaired Program acquired microcomputers, it was the responsibility of the program to implement their use. The Hearing Impaired Program supervisor and division head have provided vast support in terms of equipment, flexibility and time allotted to develop the computer services. Support has also come from computer personnel throughout the district in the form of time and expertise. The mainstream computer teacher in the same building as the Hearing Impaired Program has given of his insight, energy and cooperation on a daily basis. Fellow teachers in the Hear-Impaired Program have also made big contributions in providing their enthusiasm, time and creative ideas. In effect, the evolution of effective computer services for the hearing impaired students has come about because people whose common goal is to produce quality education for students. A great deal of time has been expended in the planning and implementation of ideas.

The approach used is comprehensive in that it involves both hearing impaired and mainstream students, their teachers, and administrative personnel. The various learning levels of the hearing impaired students are met. Interaction between hearing impaired and mainstream students and teachers is facilitated through instruction in the mainstream setting and the use of student programmers to write software designed by the hearing impaired program. Teachers benefit by having another means to provide instruction and individualize learning. The administrative and program levels benefit by having easier access to information.

The essential ingredients of a well-defined goal, wide support, enthusiasm, effective communication and flexibility have worked well for the High School District 214 Hearing Impaired Program. Continued positive growth will depend upon the continuation of these factors.

OCCUPATIONAL PROFICIENCY PREPAREDNESS
-or-
FUTURE SHOCK?

Sue Adams

Asst. Prof., Dir. of CO-OP, Computer Science Dept., TCJC-NE, Ft Worth,TX

One important question that community colleges should be asking and acting upon at the present time is, - "Are we preparing our students for Occupational Proficiency or Future Shock when they enter their career fields of computer science?"

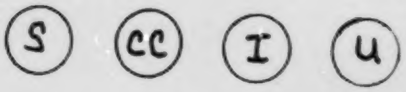
One of the fastest growing interests of all ages at the community college level is the field of computer science. It is a discipline that is so protean that no one knows the extent of its potential growth.

TCJC in Ft Worth, TX, as well as other colleges in our area are committed to excellence in all aspects of the college program. Quality education is the primary objective of all programs, whether in university parallel curricula, occupational education, or continuing education.

Since the community college is committed to excellence - one of its purposes is to prepare students in a comprehensive way of knowledge and skill to enter a field that once was considered only for the elite — the computer science field.

The synthesis of this situation is, in order to cope with the dynamic changing environment today, we need to attempt to rally our forces and get as much participation and involvement as possible from industry, the community college and university level, together thwarting the dilemma of unknowns and concentrating on the realities in the computer field of the 1980's.

The antithesis of this situation would be no communication between the entities involved.



(no communication)

S=Student CC=Community College
I=Industry U=University

Figure 1.

The student in Figure 1. above would be the ultimate loser. Industry would be frustrated with the inept product (the student) that the colleges and universities were turning out. The community college would be frustrated because their hours would not transfer to the 4-year university. The student again being the loser — an inevitable victim of loss of hours.

As a thesis of this situation let us explore several avenues of thought to bring together various techniques and strategies into a unified approach that will better enhance the community college product (the student) in the area of computer science.

According to Mr. Hollis Latimer, Chairman of the Computer Science Dept. TCJC-NE, one of the most effective tools at the disposure of the community college Computer Science Dept., is its Advisory Committee. The committee consists of members selected from an appropriate variety of business activities in the community. The Advisory Committee is a means of keeping abreast of the 'real' world in the computer field. In addition, these committees serve as communication channels which help provide the college with information necessary for making changes in curricula and suggestions for new programs and facilities.

Dr. Jim Poirot, Chairman of the Computer Science Dept., North Texas State University, Denton, Texas would like to see stronger communication ties between community colleges and universities and industry. The ultimate goal being students who come through the community colleges would not have to be so concerned with the problem of losing transfer hours and be able to concentrate on getting their education in their chosen career field of computer science.

Mr. Gary Weaver, Head of the Systems Branch - Benefit Plans Group, Mobil Oil, Dallas, Texas, believes the communication between the three areas is improving. However, he would like to see more exposure and involvement with the community colleges and universities in new facets of the computer field, such as telecommunications and technical writing. Bridging the gap with industry on new innovations that are occurring in the computer field is a must with all aspects of industry.

Clearly, we have not followed the path of Alice in the classic tale of Lewis Carroll's Alice's Adventures in Wonderland in which the following dialogue takes place . . . . .

"Cheshire-Puss,"....said Alice, "would you tell me, please, which way I ought to go from here?"

"That depends a good deal on where you want to get to," said the Cat.

"I don't much care where——" said Alice.

"Then it doesn't matter which way you go," said the Cat.

"——so long as I get somewhere," Alice added as an explanation.

"Oh, you're sure to do that," said the Cat, "if you only walk long enough."[1]

We do know where we are going and we have a path to follow. However, ..."It's a common fact, the more time you spend in planning the less time you'll need for completing a task."[2]

As a thesis to this situation, there are certain propositions to follow for completing our task. First of all, Figure 2. illustrates a possibility of a more effective flow of communication to better produce a quality product.

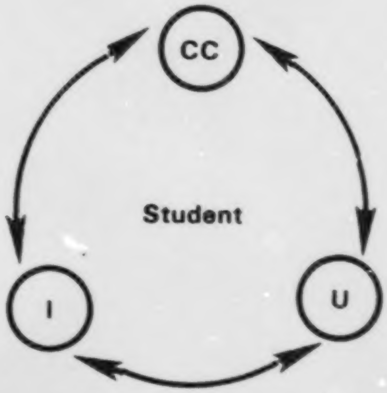


Figure 2.

The dissemination of information with freedom of flow as illustrated in Figure 2. will give more breadth and depth to the student in the long run preparing to start his/her career.

It is not the intent of this paper to condemn any aspect of the above triangle. It is intended to create awareness to the intensity of the problem facing all of us working in a field where new technology is rapidly being developed and old technology is constantly being changed and updated. Also this paper is intended to enrich existing strategy to meet the demands of our field.

More interaction is definitely needed. Career days are a good way to create a true picture of what the field 'is like. It can be specifically set up for the computer science student to give adequate exposure to what is expected on the different levels of the road before them (Community College, University then Industry or Community College then Industry). Career days should be set up in order that the students who are seriously considering computer science as a career field can become acquainted with qualified mentors from the market area.

TCJC-NE Computer Science Dept. (having approximately 2,000 students) considers computer fairs a fine opportunity to bring industry, university, and students together in an atmosphere of learning and communication.

The CO-OP program is an excellent way to tie all aspects of these entities together. It is a procedure whereby, on the community college level, an effort is made to meet the special needs of the student and of industry.

Section 171 of the Amendments to the Vocational Education Act of 1963 states:

> The Congress finds that cooperative work-study programs offer many advantages in preparing young people for employment. Through such programs meaningful work experience is combined with formal education enabling students to acquire knowledge, skills, and appropriate attitudes. Such programs remove the artificial barriers which separate work and education and, by involving educators with employers, create interaction whereby the needs and problems of both are made known. Such interaction makes it possible for occupational curricula to be revised to reflect current needs in various occupations.

Students in this type of 'internship training' serve as full duplex channels of communication from industry to education and vice versa. This type of internship training provides learning facilities that cannot be duplicated on campus and, in reality, effectively assists the college in serving the needs of the people and the community. Such service is the primary reason for the existence of any community college.

Thus far, our discussion has presented the problem, and some propositions to connect communication channels between industry and the community colleges.

Now for the most difficult task of attempting to connect the three entities together through communication channels as Figure 2. shows.

It is proposed a committee be set up in each state consisting of representatives from industry, community colleges and universities to better define guidelines as to the program of instruction needed to be developed to prepare skills and knowledge required for entry into the computer science field.

It is further proposed that attention be given to coordinating the efforts of preparing students to have a more efficient and logical transition from community colleges on to universities and ultimately into industry.

In conclusion, let us review a quote from Professor William James who wrote ages ago . . .

> "Let no youth have any anxiety about the upshot of his education, whatever the line of it may be. If he keeps faithfully busy each hour of the working day, he may safely leave the final result to itself. He can, with perfect certainty, count on waking up some fine morning to find himself one of the competent ones of his generation, in whatever pursuit he may have singled out."[3]

Our primary goal should be to produce a final product who is one of the competent ones of his/her field. Let us get on with the task before us.

LEFT-HANDED LEARNERS, THEIR HEMISPHERICITY PROBLEMS AND COMPUTERS

George Donoian, EdD **and**
Eleanor Donoian, BA

INTRODUCTION

Our purpose is to acquaint you with the learning problems affecting left-handed students, especially on the eve of an educational computing revolution. We urge you to delve into the literature at your leisure to gain a greater understanding than can be possible within the limitations of this presentation. Since we come as educators schooled in learning theory, psychology and education, we can prepare you to influence the decision makers regarding software purchases and/or development of "homemade" programs by well-intentioned colleagues hooked on educational computers. In either case, the left-handed students must be regarded as a significant minority.

THE PROBLEM IN PERSPECTIVE

Schooling is essentially left-brain development.[1] Little is done to provide affective and psychomotor domain educational experiences[2] to produce well-balanced students who really want to learn more than dates, places, people, addition, subtraction, multiplication and division. Recently Prince expressed his concern for the neglect of right brain development or the education of left-handed students and the affective training for all. He says, "Because we operate in such a sequential-seeming world and because the logical thought of the left hemisphere is so honored in our culture, we gradually damp out, devalue and disregard the input of our right hemispheres. It's not that we stop using it altogether, it just becomes less and less available to us because of established patterns."[3]

Springer and Deutch blame our educational systems which limit schooling along a narrow segment of human capabilities spectrum and by so doing deprive one half of our brains from developing and worse yet, missing out on the talents of both brain hemispheres.[4]

Sagan, of "Cosmos" fame supports the previous arguments about deprivation of the right brain hemisphere and insists that the effective pursuit of knowledge requires both hemispheres working in tandem.[5]

Our point is this: The left-handed students stand no better chance of equal educational access with computers than with human teachers unless computer software developers understand the problem and create appropriate programs to compensate for this uniqueness.

All students are shortchanged by the present curriculum imbalance. Worse yet, the left-handed ones suffer most because they are right-brained, while the cognitive instruction is directed to the opposite hemisphere. This minority is only ten percent of any school population! For the majority who have dominant left brains,[6] the cognitive educational program aimed at their advantage side is fine, but they too are deprived of right-brained development.

With the entry of the computer into the classroom, it is our contention that we will witness even less right-brained learning activities unless concerned teachers, learning specialists and educational computing experts remedy the situation. If computer coursework parallels present day classroom teaching, then nearly all the educational program content will be aimed at students' cognitive domain. However,in all this, left-handed learners remain the most disadvantaged in school, a sad commentary on modern schooling, with or without computer-assisted instruction.

THE LEFT-HANDED IN SCHOOL

We know that the greatest amount of accommodation for left-handers takes place in the early elementary grades. If there is any adjustment, it is one-way, i.e., the left-hander must make it in a right-handed setting. By the time he or she is graduating from the sixth grade, little remains of any teaching geared toward helping left-handers. However, this is not true in the beginning. All students are taught to print in what is correctly called manuscript. Care is taken by teachers to make certain that all students master this complex skill which introduces youngsters to formal language training. Later on the young scholars are taught systematic handwriting or cursive writing. Tolerating the left-handers constitutes the optimum instructional efforts rendered to the visibly handicapped who from ancient times on have been described negatively as devious, different or awkward. From the Latin word sinistra we get sinister or left-handedness as sinistral. On the other hand, pardon the pun, right-handedness is dextral which denotes rightness, appropriate and standard.

As late as pre-World War II days, some teachers tried hard to "correct" left-handed students by converting them - a heinous change when you consider that left-handers were forced to use their recessive brain hemisphere contrary to a well-established dominance which became evident in the first months of their lives. Luckily this crusading spirit does not prevail today and natural handedness is not a bone of contention with well-intentioned, but misguided teachers and parents who could have unwittingly caused cerebral disturbances.

THE EDUCATIONAL COMPUTER AND HOPE

If we are to comprehend the nature of the left-handed learner and realize how the classroom computer can be beneficial to him, we need to look at the computer carefully. The foremost advantage the computer possesses is the time factor. Educational computing is in its infancy and not yet even a toddler by any standard.

There is more hardware quality now than software of value, despite the publishers' hard-sell. We dare any software peddler to come forward with data which will support his contentions of near universal achievement by students using his materials.

What educators, computing specialists and learning theoreticians do will certainly influence the course of the computer in the school environment and improve education, but this must come about via the insiders like ourselves who will not succumb to the sales pitch of software vendors.

While the timing is absolutely essential for increasing the use of educational computers, there is a more fundamental mission. The computer is now in the best position to exploit an initiative which no other classroom technology had the opportunity to achieve. It is our judgment that *computers by their design, function and capabilities could be utilized as master teachers, hence, producing master learners*. Visit a typical classroom and witness standard instruction. More than likely, content of the lesson or test being given will be cognitive, i.e., learning to know. According to Bloom's taxonomy of educational objectives,[7] the first domain of learning deals with *cognition*. Therefore, we expect that the instruction and testing will include a wide range of facts to recall. Unfortunately, most schooling remains bogged down at this lowest level of learning. For if we are to teach beyond, we would move up to *comprehension*, grasping the meaning of information acquired or the genesis of understanding.

Level three cognition involves *application*. Students having once acquired information and understanding should be able to do something with it - the ability to use the newly acquired skills and knowledge represents the next higher cognitive skill.

Having acquired the skill in the application of knowledge, we should move students on to *analysis*, the ability to break down information into its component parts so that the organizational structure and function become apparent to the learner.

The next higher level of cognition is *synthesis* or putting parts together into a meaningful and, perhaps, new way. This is the emergence of creative behavior, but please note that we must teach students to bring them to this point.

The highest form of cognition is *evaluation* or the ability to judge the value of knowledge and skills for a given purpose. Learning outcomes which increase student evaluative abilities are at the apex of the cognitive hierarchy which requires the lower five cognitive levels, i.e., knowledge, comprehension, application, analysis and synthesis, to be mastered.

With present CAI software emulating the typical classroom model, we see a conspicuous lack of educational substance indicating a need for teachers and computer specialists to go beyond cognitive levels one and two in developing CAI materials. This is imperative.

Krathwohl's taxonomy of educational objectives of the *affective domain*[8] is the companion piece to Bloom's cognitive system. Therefore, affective learning is directed toward supporting and extending the cognitive domain by undergirding it with student interest, attitudes, values and appreciation. All these reside in the right train where aesthetic, emotional, intuitive, imaginative and creative efforts originate.

The last learning taxonomy deals with the *psychomotor domain* and requires only a brief discussion. It is apparent that a mind-body relationship is a prerequisite. Therefore, psychomotor development in the skills acquisition process arising out of need or want, and not a part of the instinctive or involuntary movements, we perceive a need or want to affect behavior. Let us take two examples to illustrate our point: first, an infant reaching for a small ball with a diameter of 1½". Seeing this bright red sphere, the child reaches for it with a whole hand grasp and brings the ball to his mouth, an elementary example. While no language was required, we are certain that the infant executed purposeful behavior. Eight years later, the same child is ready to bat a pitched ball. This complex psychomotor sequence requires the boy to swing at the ball which results in a strike. This awkward, emulative attempt is the model swing - the initial step in psychomotor learning.

CHALLENGE

Soon educational computers will be available to every student. Software should be compatible for both right-handed and left-handed students. Both hemispheres of their brains should be educated. Software writers should recognize and appreciate that each brain hemisphere has specific functions which need stimulation and integration. We think it is quite impossible for today's teachers to provide educational experiences that include cognitive, affective and psychomotor development. History does not indicate any hope. Cognitive skills involving information acquisition and some understanding seem to be the easiest, fastest and most convenient elements of so-called education to teach and hence to learn, and finally test.

Classroom computers stand at the threshold of a revolution to improve education. It will be incumbent upon people like us to effect improvement in the delivery of knowledge and skills and the **development** of values and attitudes whether they are for left- or right-handed students. No student should be left unaffected. Computers now have tremendous voice, graphics, musical and sound effects, visual and manipulative bilateral features, modeling, gaming, simulation and problem-solving potential. Are we going to take advantage of these present-day capabilities and employ them in the learning enterprise?

BIBLIOGRAPHY

1. Hunter, M., "Right Brained Kids in Left-Brained Schools" Today's Education, Washington, D.C., Nov.-Dec., 1976

2. Herron, J., Editor, Neuro - psychology of Lefthandedness, Academic Press, New York, NY, 1980

3. Springer, S.P. & Deutch, G., Left Brain, Right Brain, W.H. Freeman & Co., San Francisco, CA, 1981

4. Ibid., p. 192

5. Op. Cit., p. 193

6. Cummings, C., Teaching Makes A Difference, Teaching, Edmonds, WA, 1981

7. Bloom, B., et al, Taxonony of Educational Objectives Handbook I: Cognitive Domain, D. McKay Co., Inc., New York, NY, 1956

8. Krathwohl, D., et al, Taxonomy of Educational Objectives Handbook II: Affective Domain, D. McKay Co., Inc., New York, NY, 1956

BIG PROBLEM - STUDENT SCHEDULING AND GRADE REPORTING
BIG SOLUTION - THE MICROCOMPUTER

by

Dr. David K. Mosow
Roosevelt Schools
Phoenix, Arizona

Ms. Brenda Bolton
and
Mr. Henderson Young
Davidson High School
Mobile, Alabama

There are several problems common to a majority of middle and secondary school administrators. Many of those problems seem to cluster around the topic - Student Accounting. A large portion of the personnel and fiscal resources available to the building administrator must be concentrated on student accounting. Although sometimes unwritten, it is generally assumed that the building administrator can accurately locate a student when necessary, change demographic information for students and teachers, change schedules for students and teachers, inform others of the changes, and accurately accumulate and report student grades, grade point averages and class standings.

In many districts this task is deemed too large for the individual school to handle. Thus, these tasks are relegated to a large centralized computer installation. These installations represent service bureaus or district office operations. In the past the concept of distributed data processing was too costly to justify for most school systems. Recent technological developments though have allowed for a tremendous amount of computing power to be available at a fraction of large installation costs. This premise lead Davidson High School, Mobile, Alabama, to investigate the utilization of modern technology to solve several pupil accounting problems.

The school administrators began to research the feasibility of using a microcomputer to perform these tasks. They eventually talked with computer applications specialists Steve Olensky and Dr. David Mosow. This twosome began researching available student accounting software for the microcomputer. Since very little software was available, the application specialists elected to develop a computer program designed to be as generic as possible and still meet the individual needs of Davidson High School. The Apple II Plus microcomputer was selected since it is well supported and likely to be found in many schools. The applications specialists along with Ms. Brenda Bolton and Mr. Henderson Young from Davidson comprised the Application Team. The following discussion describes the problems and needs from Davidson and the solutions developed by the team.

PROBLEM ONE:
Big School-Small Computer

It became evident that it would be quite difficult to develop intricate programs to handle the school enrollment (1700) and track demographic information. Either little information would be monitored or a larger storage device would have to be used. The team decided on a large storage device and selected a 10Mb hard disk from Corvus Systems to interface with an Apple II microcomputer. In effect Davidson High School now had a large computer with the latitude necessary to develop a scheduling program (later entitled Student Scheduling and Tracking System SSTS).

PROBLEM TWO:
What To Collect

One of the most difficult tasks involved foresight which can be quite fickle when projecting future computer applications. The Application Team needed to insure that all student information was collected that could be used for present applications and future endeavors (an impossible task). The team looked at other office tasks and decided on the following demographic information:

Student Number
Student Name
Date of Birth
Grade
Race
Age
Address

Father's Name
Father's Address   Employer

Mother's Name
Mother's Address   Employer

Emergency Contact
Emergency Phone   Doctor

Bus Number

This set of variables was included for each student. Some of the information is not presently being used, but in the future, information about bus ridership, race by grade, and emergency information will be available for analysis or retrieval.

PROBLEM THREE:
Batch or Keyboard Entry

The Application Team wrestled with this concern for several weeks and finally revised the question. The real issue was--How Clean Must the Data Base Be? The answer became obvious. The data base must not contain invalid course numbers, teacher numbers or student numbers. Thus, keyboard entry was selected since the program could immediately search the data files and insure validity or not allow invalid information to be written in those files. Using a scanner and mark sense sheets would constitute a batch mode and the school would increase the probability of corrupting the data files by loosing the "Up Front" error checking capabilities. It was also decided that it actually took longer in total person hours to bubble mark sense sheets and have them read by a scanner than simply to enter the information from the keyboard.

PROBLEM FOUR:
Quick Update

The Application Team also determined that quick access to information and update of all student information was a must. Logically the team decided that the system had to be quicker and more accurate then the present non-computerized system. To address this problem, routines were built into the system to allow quick demographic information change, schedule change and reprint of the schedule, grade change and reprint of grade cards, and master schedule information change.

There were several other problems which were addressed during the development of the system; however, the aforementioned concerns were the important considerations. Analyzing these concerns "Up Front" was instrumental to insure a successful development phase.

OPERATIONALIZING

The system became operational in several stages. Initially, the student demographic portion of the system was brought on line. As the Application Team worked on the scheduling and update phases, the school staff entered demographic information for all students. That made way for scheduling. Student course requests were entered and the scheduler was initiated (See Exhibit One). During this time a parallel (back up) system was also activated. The central office main frame computer was also scheduling the school in case SSTS was not ready for the school year. Fortunately SSTS worked in time to utilize the scheduling run for student schedules. The school staff concentrated on changing schedules during the first weeks of the academic year while the Application Team developed grade reporting routines (See Exhibit Two). The last step was to finalize all menus and tie all components together so little training would be needed to operate the system.

FINAL ANALYSIS

In retrospect the project has been an unqualified success. The school staff, administrators, teachers, and clerical personnel quickly embraced the concept since the time and personnel savings have been dramatic.

As important however, is the quality of information now available to school staff and students. Originally it was quite difficult to locate a student after school started since class changes invalidated the hand processed locator information. Students naturally took advantage of the information void and would drop classes they were struggling in and never report to the replacement classes. The teachers never knew to expect the students and the office never knew there was a problem. The computerized system frequently allows for new rosters to be printed and sent to teachers for verification. The integrity of the locator information is preserved and those students attempting to take advantage of the system are discovering that things have changed. Future plans involve a daily report of drops and adds to be available for teachers. This option may be valuable during peak schedule changing times (i. e. the beginning of each semester).

Secondly, grade reporting was a monumental task in the non-computerized system. Each teacher would have to determine the homeroom for each student on their class rolls. A grade for each student was transmitted to the proper homeroom. The homeroom teachers then transcribed that information onto the student's report card. The homeroom teacher also transcribed the grade information onto an office copy and transmitted that to the office for posting on the permanent record cards. That means approximately 34,000 grades were written, rewritten or copied at each grade reporting time. As might be expected both the error rate and the confusion level were high.

The new system is much simpler. A roster is generated for each class and distributed to the proper teachers. The teacher simply places the proper grade for that particular course next to each student's name and returns the grade sheets to the office. The computer operator then enters the grades from the roster sheets via the computer keyboard. Grade entering for a teacher's five classes takes on the average six to ten minutes. The decrease in grade handling drastically decreased the error ratio. Less than a dozen human errors were detected during the first reporting period. There were no program errors. During the second reporting period, the staff uncovered a dozen coding errors on the part of the teachers and one keyboard mistake out of 32,000 entries. When one of these errors is discovered or an incomplete grade is changed, that student's grades are called to the screen and the proper changes are made. A new report card is then printed and the old one collected and destroyed. No student can have a report card with a hand written or typed-over correction. This fact alone insures grade integrity.

One of the administrators at Davidson High School conducted a time study to compare time spent on the recording of grades over and above that required to figure grades for students. Using the manual system teachers averaged 150 hours a year to record, transmit, and rerecord grades above and beyond that required to figure grades. With SSTS each teacher spends approximately four (4) hours per year above and beyond that normal time required to figure grades. The amount of time saved has been much more than expected.

In summary, the project is basically complete. Naturally there will always be changes. The bottom line is - IT'S WORKING AND IT'S AFFORDABLE TO MANY SCHOOLS. In an age of increased information requirements to manage available resources, that's important.

```
0888855285  WAYNE [redacted]                     1201

COURSE #  DESCRIPTION      TEACHER     ROOM S P
-----------------------------------------------
152500-01 PSYCHOLOGY       GARRETT M.  8    1 1
152500-01 PSYCHOLOGY       GARRETT M.  8    2 1
221750-01 MARINE SCIENCE   KULAKOWSKI  114  1 2
221750-01 MARINE SCIENCE   KULAKOWSKI  114  2 2
152250-02 AMERICAN GOVT.   MOORE J.    9    1 3
152260-02 ECONOMICS        MOORE J.    9    2 3
112110-05 ENGLISH 12 COMP  SEXTON V.   105B 1 4
112130-05 ENGLISH 12 LIT   SEXTON V.   105B 2 4
250300-02 GEN ALG I PT 2   DAVIS N.    14   1 5
250300-02 GEN ALG I PT 2   DAVIS N.    14   2 5
440200-01 *VARSITY ATH PE  MARSH R.    GYM  1 6
440200-01 *VARSITY ATH PE  MARSH R.    GYM  2 6
```

```
                     MOBILE COUNTY PUBLIC SCHOOLS
                  W.P. DAVIDSON HIGH SCHOOL  1981-82
                         SECTION: 1201/2                  DATE: 02-01-82
0888864540  [redacted]
    QUARTER 1              QUARTER 2                          SEM   SEM          CREDITS
    SUBJECT        GRADE   SUBJECT         GRADE  CONDUCT     EXAM  GRADE        EARNED

1.  *CHEMISTRY II    B   *ADV ENGLISH 12 A    A              X     A             0.5
2.  PHYSICS          A   INTRO ANALYSIS  B    A              X     B             0.5
3.  *ADV ENGLISH 12  B   *ADV ECONOMICS  B    A              X     A             0.5
4.  INTRO ANALYSIS   B   *BIOLOGY II     A    A              X     B             0.5
5.  *BIOLOGY II      B   *CHEMISTRY II   A    A              X     B             0.5
6.  *ADV ECONOMICS   A   PHYSICS         C    A              X     B             0.5

SEM GPA: 3.7   CUM GPA:       RANK:       DAYS ON ROLL:       DAYS ABSENT:

A-EXCELLENT B-GOOD C-AVERAGE D-POOR E-FAILING I-INCOMPLETE N-NO CREDIT
S-SATISFACTORY U-UNSATISFACTORY

                           PARENT SIGNATURE: _________________
```

APPLIED DISCRIMINANT MODEL FOR INFORMATION SYSTEMS EVALUATION

Avi Rushinek
Sara F. Rushinek
Joel D. Stutz
University of Miami

ABSTRACT

This paper describes the construction and the use of a discriminant model to evaluate information systems which are implemented on an experimental basis. The discriminant model is constructed from evaluation survey data. Discriminate scores are generated by the discriminant function prior to the use of the new information system and subsequently to the implementation of this information system. The differences between the prior and the subsequent discriminant scores are then used to indicate the marginal effects of the newly installed information system.

INTRODUCTION

Most of the current research on computer assisted instruction (CAI) does not consider extensively the relationships among CAI, students' attitudes and instructors' attitudes. These relationships may be very important since the instructor often is the only one who decides whether or not to use a certain CAI program as a supplement to classroom instruction. The instructor may decide to use CAI if it is expected to improve his or her evaluation by the students. Further, students' attitudes toward the instructor may affect their learning process (Crawford, 1970).

The existing research of instruction supplemented by CAI is rather extensive. However, most of these studies fail to examine a critical variable, namely, the effect of CAI on the instructor and course evaluation. Romaniuk and Montgomerie (1978) studied the use of CAI as a supplement to a BASIC introductory computer course taken by first year computer systems students. Romaniuk and Montgomerie compared the performance and the time required to complete the course. However, they did not investigate the effect on the instructor and course evaluation due to the use of CAI. The present study attempts to clarify the picture by investigating the impact of CAI on Course Instruction Survey.

A similar pattern is apparent in a study performed by Hawkins (1977). The main purpose of Hawkins' survey is to help improve the management, development, and implementation of the use of computer facilities in education. The general recommendation of the Hawkins' study emerges as a need for rethinking the organization, the financing, and goal-setting of development projects in the field of CAI. Evaluation of the changes in the Course Instruction Survey due to the implementation of CAI is a way of rethinking this issue.

METHODOLOGY AND PROCEDURES

The subjects in this study were students of a major southwest university in the College of Business Administration who participated in an introductory course of Information Systems. Both classes were taught by the same instructor. The Fall class was designated at random as an experimental group and thus was exposed to CAI programs which taught the BASIC computer programming language. These CAI programs were used as the experimental treatment given only to the experimental group as homework. The Spring class, the control group, was not permitted to use these tutorials.

This study was designed as a pre/post test with an experimental and control group. The pretest was administered at the midterm, prior to the treatment (CAI) period, which started after the midterm. The pretest included a comparison between the experimental and the control groups on students' attitudes towards the instructor and the computer, as well as students' midterm exam scores, GPA and SAT score. The pretest showed no significant difference between the two groups. Thus, any differences can be attributable to the CAI treatment.

The samples used in this study consisted of 93 students in the control group and 146 students in the experimental group. The experimental group students were required to run the tutorials as a homework assignment accounting for 5 percent of the total grade for the course.

A time schedule for the completion of the CAI tutorial assignments were recommended to the students, however, the students were allowed to complete the assignment after the suggested due-date. Students were able to repeat a single assignment an unlimited number of times. Credit for each of these assignments was based upon the highest score that students received from the tutorials.

HYPOTHESES

It is hypothesized that there will be overall differences between the control and the experimental groups' scores such that:

(A) The experimental group will express a significantly more positive attitude toward the instructor, therefore, their attitudinal score on the variable entitled "Instructor" will be lower than the control group.

DISCRIMINANT ANALYSIS COMPARISONS BETWEEN THE GROUPS

The statistical theory of discriminant analysis assumes that the discriminating variables have a multivariate normal distribution and that they have equal variance-covariance matrices within each class. However, this technique is rather robust and these assumptions need not be strongly adhered to.

Since the discriminative variables are usually coded in terms of raw scores of semantic scale in the Course Instruction Survey, an unstandardized discriminant function may be required in order to compute the evaluation change. The unstandardized coefficients in Table 1 have to be multiplied by the raw values of the evaluation variables to arrive at a discriminant score. Accordingly, the means of the non-CAI and the CAI classes of the variable 1 should be multiplied by .503245. After adding the products of the means and their coefficients, the constant -1.2533 should then be added to adjust for the grand means and thus the discriminant score is obtained. One may observe that the unstandardized coefficients do not report the relative importance of variables since they have not been adjusted for the other variables within the discriminant function.

TABLE 1

UNSTANDARDIZED DISCRIMINANT FUNCTION COEFFICIENTS

Item		Coefficient
Variables	1	.503245
	2	.235592
	3	-.804346E-01
	4	-.120808
	5	-.991771E-01
	6	.745922E-01
	7	-.164727
	8	.491312
	9	-.275500E-01
	10	.938897E-01
	11	-.220868

Item	Coefficient
12	-.117298
13	.438470
14	.244223
15	.323780E-01
16	-.281542
18	.803288E-01
19	-.217143
20	-.251526
21	.106318
Constant	-1.25330

By averaging the scores for the cases within each group, with the experimental and the control groups, we arrive at a group mean on the aforementioned discriminant function. For a single group, the control group (Contr Gr) for instance, the means of all the functions are referred to as the group centroid, .74605 in the plot. This is the most typical location of a case from the control group in the discriminant function space. A comparison between the group centroids tells how far apart are the group's centroid in the reduced space. The range of raw scores 1 through 7, is replaced by the range of -3.000 to +3.000, expressed in terms of standard deviation in the plot. These units may be helpful in converting the difference to percentages of the normal distribution.

Examination of the plot describes the separation of the group centroids and their relative locations. One may observe the overlapping region is small (1.00 through 1.52) relative to the total range (-3.00 through +3.00). This means that the discriminating variables are very successful in discriminating. This may be further clarified as one observes the overall statistical significance in Table 2. This overall adequacy of the discriminant function may be evaluated empirically by using this discriminant function. Such an empirical test of the discriminant function will be performed by classifying the cases, used to derive this function, into the control group or the experimental group comparing predicted group and comparing predicted group membership with actual group membership. The following in Table 2 are the results of such a comparison.

TABLE 2

OVERALL SIGNIFICANCE TEST FOR THE DISCRIMINANT FUNCTION-PREDICTION RESULTS

Actual Item	Group Code	No. of Cases	Predicted Gr Membership Contr Group	Exper Group
Control Group	1	93	67. 72.0 %	26. 28.0 %
Exper Group	2	124	26. 21.0 %	98. 79.0 %

76.0 percent of known cases correctly classified.
Chi-Square = 58.843
Significance = .000

Table 2 shows that 72.0 percent (PCT) of the members of the control group (Contr Gr) were correctly predicted to be in the control group (Contr Gr) by this discriminant function. Experimental group (Exper Gr) is predicted with a higher precision of 79 percent, which yields an overall 76.0 percent of known cases correctly classified. Finally, chi-square of 58.843 with a significance of .000 indicates that we may infer statistically from this sample to the population of electronic data processing (EDP) classes. Therefore, we may use this aforementioned discriminant function for determining the change in instructors' evaluations due to the use of CAI as a supplement to the teaching of EDP.

It may be seen that there is a significant difference between the group's centroid. The experimental groups' centroid of -.55954 in the plot is much smaller than the control group centroid of .74605. This indicates a more positive attitude by the experimental group since the attitude and the centroids scores are negatively related. This confirms the hypothesis that the experimental group has a more positive attitude due to the CAI and that this discriminant function is more efficient in deriving these conclusions in comparison with the T-test since it does not require additional statistical inference, once proven to be significant. Accordingly, the discriminant score can be reported to the instructor together with the results of the course instruction survey to be used by the instructor. Although the T-test of the variable Course may indicate no significant improvement, when lumped together with the variable Instructor, by the discriminant analysis, it turned out to be rather significant. The latter takes into consideration the interaction between the instructor and the course he teaches and therefore may be preferred to a T-test of paired comparisons.

This paper indicates that a positive relationship may exist between the students' attitude toward the CAI and their evaluation of the instructor and the course. Therefore, a modification of the CAI programs that will improve students' attitudes toward the CAI, may also improve the instructor and course evaluation. Subsequent research could investigate ways of modifying a CAI system and presents different methods of evaluating the results of such modifications.

CONCLUSIONS

A number of conclusions can be drawn from the analyses presented here. First it can be concluded that CAI used as a supplement to classroom instruction has a statistically significant impact upon students' attitudes toward the instructor. It was found that the changes in students' attitudes are positively related to the evaluation of the CAI used. That is to say, the experimental group expressed better attitudes than the control group. Students evaluated positively the CAI they used.

If a CAI program is known to improve students' evaluation of their instructors it may become more attractive to instructors and more extensively used.

In conclusion to this study, it is suggested that supplementing accounting instruction with CAI facilities can be both effective and efficient in promoting educational objectives, including improvements in students' attitudes. The results may encourage the expansion of resources devoted to CAI as a tool for achieving desirable educational objectives.

REFERENCES AVAILABLE UPON REQUEST

"The Computer in the Schools -- What's Been Said to the Teacher?"

Martin Saltz
West Virginia University
506 Allen Hall
Morgantown, WV 26506

Input. Output. Micro. Mini. Program. BASIC. CRT. Coursewriter. Architecture. Peripheral. Floppy disc. Plotter. PLATO. Reader. Printer. Hardware. Software. Escape. Special character. Communications. Interface. Modem. Hard copy. Loop. Subroutine. Function. Courseware. Intelligent terminal.

How many classroom teachers know what these terms mean in the context of today's technology?

Comparatively few classroom teachers are conversant with the vocabulary of the computer era, let alone with the computer itself and computer applications to the teaching of elementary and secondary school subjects.

This researcher decided to examine selected professional literature available to elementary and secondary classroom teachers, and to make suggestions intended to aid in developing "computer literacy" for the classroom teacher.

For most classroom teachers, the computer field is relatively new. Opportunities for classroom teachers to examine computers and their educational applications vary greatly according to such factors as:

* the educational discipline
* the school system's budget
* the school system's philosophy
* the teacher's past training
* the individual's attitude towards computers.

Until very recently, secondary school teachers have been able to acquire computer knowledge more readily than elementary school teachers. Considering the historical development of computers, this should come as little surprise to the reader. Further, the computer itself usually was more available to mathematics and science teachers than to other teachers. The microcomputer and word processing machines have changed the picture, however. There is an ever-increasing amount of computer-related information being directed towards elementary school teachers and to teachers in subject areas other than mathematics and science.

Where can teachers turn for information and instruction? In-service programs can be developed. Teachers can purchase books on computers and computing. Teachers can read magazines. Courses can be added to teacher-training programs. (Enough added computer instruction and courses might even tempt present and future teachers to seek career changes.)

In early February, 1982, an ERIC search was conducted using the search parameters:

1. 16629 COMPUTER? *MICROCOMPUTER

2. 51669 ELEMENTARY *SCHOOL

3. 1127 2 and 1

Over 1120 references relating ELEMENTARY *SCHOOL (2) and COMPUTER *MICROCOMPUTER (1) were identified as having issued between 1964 and 1981. These included:

* articles from professional journals and popular journals

* surveys from governmental and private organizations

* reports from state education departments and city school districts

* documents from universities, councils and committies

* books (hundreds of pages in length)

* articles (2 - 3 pages in length)

* individuals speaking and groups reporting.

(Note that this was without searching SECONDARY *SCHOOL and COMPUTER +MICROCOMPUTER.)

The writer will involve the session participants in discussing and augmenting selected summaries of the available literature. Working together, participants will develop a method for sifting, selecting and evaluating the current literature.

... Where can classroom teachers, just now becoming aware of computers, go to find introductory material? Where can they find material that will not bog them down in vocabulary and concepts better suited to computer technicians?

... What educational journals appear to be taking the lead in publishing computer information directed toward the teacher-user rather than the computer professional?

... What educational journals might have an audience that would benefit from computer-oriented articles, but have not yet moved in this direction?

... What organization issue publications concerned with special educational uses for computers?

... Are there subject-oriented publications that develop some type of sequence with their articles, or is material coming to the reader in a "hit or miss" manner?

Answers to such questions are needed by teachers/users. Answers to such questions are needed by writers/producers.

As the group examines available data, the session participants will consider such issues as:

... What teacher populations are presently not being offered computer information in their field and at their level of competence?

... In what ways can teachers become involved in working with people developing the materials mentioned in the various informational banks?

... In what ways can writers best deliver the information they possess?

... In what ways can the information producers move information users from their present position, of little or no computer knowledge, to computer awareness, from computer awareness to computer proficiency, and to whatever lies beyond?

Martin Saltz
Reading Center
506 Allen Hall
West Virginia University
Morgantown, WV 26506

or: 900 Dudley Ave
Morgantown, WV 26505

Motivating underachievers--it's one of the most frustrating tasks a teacher faces. How do we convince our students that what they do--or don't do--in school today affects what they'll do tomorrow in the workplace?

As a teacher at Paxton School in Walton County, I found one solution--demonstrating a direct relationship between course work and earnings by letting students know that today's skills can pay tomorrow's bills for "wheels," "freedom," "partying," or just the costs of everyday living.

The big question to ask your students is, "What work are you going to do to pay your bills when you finish school?" Making students face this question--making them come to grips with the realities of their own future--provides a crucial link between specific skills and life after school.

Once students are convinced that they will have to support themselves--that no one else is going to pay for rent and food--they must choose a career direction. And effective career development is one of the most difficult and complex challenges facing them.

Career planning must be realistic, yet futuristic; individualistic; yet informed; encompassing, yet efficient.

Today's students are entering a work world exploding with its number and variety of new occupations--particularly in the high-technology and service areas. New workers need the traditional basic skills as well as modern technological know-how. In addition, they must be prepared to change jobs and careers as technology changes.

No teacher, no counselor, no occupational specialist, regardless of talent or dedication, can possibly fulfill all the career-decision needs of students. Fortunately, Florida now has a counseling resource that can help bridge the barriers of insufficient occupational information, individual assessment, and job-matching abilities. This resource, appropriately enough, is called "FLORIDA CHOICES."

Guidance by Computer

FLORIDA CHOICES uses computer technology to make millions of "bits" of career and educational information accessible to students and to guide them through a series of career-related self-assessments. Thus, CHOICES is referred to as both a career information system and a guidance system.

"Systems, computers, self-assessments, and more," you may be saying to yourself, "are too complicated, impersonal, time-consuming, and expensive for me or my students." Well, that's not true. FLORIDA CHOICES is really not very complicated, impersonal, time-consuming, or expensive.

Let's consider the charge of "complicated" first. The student uses a terminal that looks like a typewriter hooked to a telephone line. The terminal prints questions and answers on regular paper in simple, easily read (about fifth to sixth grade level) English (soon to include two other languages). The most difficult typing required is the student's first name. From that point on, the computer refers to the student by first name in a conversational mode. (So much for the charge that CHOICES is "impersonal.")

A student named Diane who was using the Explore route of CHOICES would get the following dialogue:

> Okay, Diane, there are a number of ways we can look for careers which may be right for you. In all of them, however, I will be asking you to answer some questions about yourself. Please choose your answers very carefully.
>
> Would you like to see the list of topics you can use in Explore? ("Y" or "N").

Diane would then respond with a "Y" or "N" (or a "yes" or "no", or a "yes" or "nope") and the program would continue.

Notice that the student is controlling where the program goes and what it does. The student and program "interact" directly; the student is "talking" to a machine, a machine that leads him/her through a flexible career exploration process, which s(he) controls and changes to meet his/her individual needs.

Routes

Explore is one of five ways (routes) to get information from CHOICES. The others are Specific, Compare, Relate, and Job Bank.

Explore allows a student to systematically examine things about himself/herself ("topics") and eliminate, by preference or ability, various careers. The 12 CHOICES "topics" are interest factors, aptitude factors, temperament factors, education level, working conditions, future outlook, earnings hours of work/travel, physical demands, physical activities, indoor/outdoor, and career fields.

Explore addresses the student's need to relate his/her individual preferences to actual jobs; to "try on" the various career options CHOICES suggests; to realistically assess his or her ability to "fit" within certain career constraints; and to learn the basis of reasonable career decision-making.

The student can monitor directly the effects of self-assessments and preferences as the machine (in a realistic fashion) poses questions, eliminates careers that do not meet those criteria, and provides a continuous tally of the number of careers still available (based on the student's own criteria).

The student may (and indeed should be encouraged to) change his/her answers at any time. This allows for the testing of those preferences falling in the "gray" areas and for determining the consequences of contemplated career development decisions.

We should not be misled into thinking that value structures in young people (particularly work-related values) are set in concrete; rather, they remain in a state of flux throughout adult life as living situations change, individual growth occurs, and emotional needs evolve. Thus, students need to learn the processes necessary to value their preferences, to determine personal factors to consider, and to mesh personal job expectations with real-life situations. Guidance systems concentrate on teaching the processes of decision making. CHOICES uses the Explore route to provide this guidance information.

Information System

In addition to serving as a guidance tool, FLORIDA CHOICES also functions as an occupational information system. It contains detailed information about some 1,000 careers found in Florida. The data available about these jobs is either United States or Florida-based (depending on the location of information requested) and includes information on several factors (salaries, supply/demand, etc.).

The Florida Occupational Information Coordinating Committee (representing the Florida Divisions of Vocational Education, Employment and Training, Employment Security, and Economic Development, as well as the Florida Office of Vocational Rehabilitation) provides support for the data updating process.

FLORIDA CHOICES is a Florida-based system; thus the system is committed to providing the most current and accurate U.S. and Florida Career information available.

Information can be retrieved instantly for any occupation, using either the Specific route (one occupation) or the Compare route (two or three occupations simultaneously). This information may either be limited (like wage only) or very comprehensive with many areas of information (like interests, aptitudes, educational level, similar occupations, etc.). Compare also allows the user to see very easily how one occupation varies from another, topic area by topic area.

Of the 1,000 or so occupations contained in CHOICES, approximately 20 percent require a baccalaureate or higher degree, 60 percent represent skilled or semi-skilled careers, and

20 percent are generally of the unskilled type. These 1,000 occupations are cross-referenced to an additional 3,000 similar occupations.

While FLORIDA CHOICES is classified as both a guidance and a career information system, it also has two other novel routes. The first of these is Relate, which is applicable to students or adults with some substantial work experience. Relate allows the user to examine previously held jobs, choose those characteristics about those jobs s(he) liked, and then obtain a listing of related occupations.

For example, a person might say, "I really like working outside but I'm tired of being a landscape gardener. I like working by myself and having a job with a good deal of physical activity."

CHOICES would ask him/her to enter the index number for landscape gardener (5515-217) and the topics s(he) wants to use (outside, working by myself, physical activity, etc.). Based on that input, CHOICES would create a list of jobs that were related to landscape gardener by having the characteristics of outside work, working alone, and similar physical activities. Related is a unique and powerful tool for job market re-entrants.

Job Bank

The fifth and newest FLORIDA CHOICES route is Job Bank. Computerized Job Bank information is provided by the Florida State Employment Service (FSES) each morning to the CHOICES computer. A student can have immediate access to some 9,000 job openings each day.

Using three sets of decision, the student specifies the type and location of the job s(he) desires. The first decision is whether to look at public (state and federal) or private (business and industry) sector jobs. Then the user must determine whether the search is to be statewide, regional, or local (city/FSES office).

Finally, either a specific occupation or a group of related occupations (career fields) can be printed out.

The printout gives DOT code, employer job title, job order number (for use with FSES), employer requirements for experience and education, minimum starting wage, and pay period. The first list contains a maximum of 30 job openings, with the option to retrieve a total of 90 using the original set of decisions.

Of course the student can change any decision to create new listings. For example, a student who always wanted to remain in his/her hometown may find that the specific occupation s(he) wants locally is not available through Job Bank. However, that job is available in the region or state. Another discovery s(he) makes is that there are locally available related jobs (career fields) that share many of the same skill requirements and other characteristics. Now the student must consider whether to stay in town and work in a related occupation or leave town to work in a specific field. The student has some information from which to make an informed decision, and can apply through the local school's counselor or occupational/placement specialist to the local FSES office for referral to various prospective employers, locally or throughout the state.

FLORIDA CHOICES, with its five routes, twelve topics and millions of bits of data, represents a very flexible system that is directly usable today by students in a variety of Florida high schools, vocational/technical centers, community colleges, universities, employment and training agencies (CETA), and vocational rehabilitation facilities.

The CHOICES Experience

For all students, the "CHOICES" experience has three phases: (1) orientation and valuing with the Guidebook; (2) interaction time with the computer terminal; and (3) follow-up counseling using the printout. The third phase is the most critical, because it is the decision-making and action phase. It is also the most difficult for students, teachers, advisors, and parents, since it requires closure. At this point decisions must be made and directions set to increase career flexibility, to open, rather than shut, doors to working opportunities. CHOICES can provide some elements to begin this counseling phase, but a human must interpret, probe, relate, and give meaning and direction to the student.

CHOICES, of course, is only a tool. Its real effectiveness is determined by human factors controlling and fostering its use. Evaluation after evaluation of every computerized career guidance or information system turns up the same finding: the measured effectiveness of the system at a particular location (school, college, or whatever) is directly correlated to the measured effectiveness of the human delivery agent (teacher, counselor, occupational specialist, interviewer, etc.). So FLORIDA CHOICES is a tool that can be extremely useful in facilitating the individual student's career development process, but it is still only one element of a process.

At the same time, FLORIDA CHOICES does provide greater efficiency for counselors and occupational specialists. By taking assessment and information-providing tasks over for counselors and occupational specialists. By taking assessment and information-providing tasks over for counselors, the system frees counseling time. Although a counselor may spend less time with each individual student, s(he) can serve more students more effectively with the aid of FLORIDA CHOICES.

The FLORIDA CHOICES system is available to every school, school district, community college, university, vocational rehabilitation facility, and employment and training agency in the state through the Center for Career Development Services. The package covers everything from computer time to a CHOICES button for your lapel.

Included in the FLORIDA CHOICES system are:

- CHOICES terminal, maintenance, paper, and ribbons.
- Two data sets.
- Telephone lines.
- Consumable Guidebooks and other student materials.
- A wide variety of counselor materials, including posters and brochures.
- A total inservice package.
- On-site technical assistance.
- Toll-free WATS line to report trouble.
- Computer time 24 hours a day, seven days a week.

If your school district or community college already has computing capability, we can often tie into it, reducing your monthly cost. While the cost of FLORIDA CHOICES depends on several other variables, the CHOICES Policy Board (the interagency group that guides the program) has determined that at no time shall the total cost of the service exceed $500 per month (excluding installation), regardless of location.

To defray costs, grants are available, and many Florida schools have applied for and received funding to pay for the system. For further information, contact the Center for Career Development Services.

Glenn Thomas is the director of FLORIDA CHOICES.

- Glenn Thomas
William Wooley

416

MICROCOMPUTER APPLICATIONS IN THE SPECIAL EDUCATION CLASSROOM

Greg Turner

Data Systems Consultant, Wayne
County Intermediate School
District, Michigan

The following is a report on Wayne County Intermediate's involvement over the past six years in the field of computer-assisted Special Education, in particular, the area of communication aids for the non-speaking Special Education student.

In 1976, WCISD contracted Michigan State University's Artificial Language Laboratory to attempt to develop portable devices for the non-speaking. This work was funded from TitleI federal funds in the spirit of Public Law 94-142, which mandates that each student shall have equal opportunity for access to mainstream education. Seven severely-mentally impaired and trainable-mentally impaired students were selected as a target group for this program, on the basis of testimony from their speech therapists and teachers that they showed significant skills in the area of language reception, that is, they could understand language directed to them. It was thought that these students were mentally capable of expressing language had they only the physical means to do so. These students were all wheelchair bound, largely diagnosed as cerebral palsied, and with moderate to severe physical impairment, including total or near total impairment of the ability to speak.

The project was pursued on two fronts: the technical, conducted at MSU with periodic visitations to the classroom site, and the speech therapeutic, which involvedthe implementation by the project's speech therapists of the so-called "Blissymbolics" pointing language system. By the end of the second year, the majority of the students were using 200 to 400 item "Blissboards".

On the technical front, MSU developed a minicomputer system and several table-top microcomputer prototypes. These unfortunately had only limited applicability in the ISD program. However, this work did lay the groundwork for later efforts and indeed must be regarded as the necessarily-prior research phase of the project.

In 1978, financial considerations led to WCISD terminating the contractual research arrangment with MSU, and it created the position of Data Systems Consultant for Special Education with the intention of bringing the technical effort in-house. Over the next two years various prototype devices were introduced into the classroom. Tutorial-style conversation sessions were conducted with these equipment to try and reinforce patterned and spontaneous communication and to encourage peer group interaction. The devices during this phase attempted to establish for the students which of several possible "input mechanisms" was most appropriate, in terms of their physical and cognitive capablities.

In 1980, the students were outfitted each with an individualized device which they then used daily in the classroom for both basic communication and for instructional purposes, which included spelling and reading, simple arithmetic, and social studies. These seven devices were individualized to each student in terms of the students' motoric and mental abilities for "input". Two students use a large 300 square touchpanel, one a smaller touchpanel arrangement, two the so-called "optical head pointing system," one a single-switch controlled scanning device, and the last a single-switch Morse Code arragnement that he keys with his foot. Each input unit shares a single enclosure with the visual electronic display, miniature printer, and speaker from the voice synthesizer.. Also common to the units is the control box which contains the microcomputer and associated electronics, power supply components, and voice synthesizer. These two boxes are interconnected by ribbon cable.

In fall 1981, these students have returned to their local school districts, with their devices: three are in POHI (physically or other health impaired, i.e. no mental impairment classification) rooms, and the remaining four are in a trainble center two of whom are classified educable-mentally impaired. These are noteworthy gains in classification, and it is apparently quite unusual to see upward reclassification of this sort. It must be pointed out that these gains must be attributed to the total program of speech therapy, remedial instruction, and device use.

Work commenced in summer of 1981 on an advanced model communication aid, which builds and improves upon the proceeding efforts. Those previous devices were not wheelchair portable, which severely limited their utility communication-wise to the students. This deficiency has been overcome in the new device. By building the 6502-based microcomputer and associated electronics in its entirety, a number of power saving features have been incorporated that permit the unit to run on regular, flashlight size batteries. Furthermore, the latest speech synthesizer and display technologies have been used which has permitted a considerable decrease in size, weight, and cost: from 40 to 8lbs, and from a component cost of $3600 to under $1000.

This unit consists of a 300 square touchpanel, single chip speech synthesizer and calculator -sized "Memowriter" for print and display, all integrated by the in-house built microcomputer. The device is powered from a stack of nicad rechargeable batteries, for a charge-to-charge life of over eight hours. For wheelchair users, the device is mounted laptray style on the arms of the wheelchair; for ambulatory users, such as an autistic student, the device will be fitted with a handle for carrying.

Besides this portable system a "second generation" table top system was also developed in summer 81. This device is intended to serve the needs of the speech therapist who wishes to diagnose students for their aptitude as regards use of a portable device, and for training the students thus diagnosed in advance of their receiving a dedicated portable system. This system consists of a 300 square touchpanel as input through a Wayne County built interface board to an Apple II computer.

Output is to a speech synthesizer module, the Apple monitor, and to a printer. Specially-written software provides the non-computer professional with a high level operating system for creation and editing of vocabulary files, which are stored on disk, one or several to each student being serviced. Operating the regular microcomputer keyboard, the speech therapist can easily and quickly create new vocabularies for the touchpanel, representing these by pictures, drawings, symbols, words, etc on a paper overlay on the touchpanel. The speech therapist can thus readily experiment with vocabulary and its arrangement on the overlay to achieve an optimal match to the student's needs. It is currently being used in an autistic program, and the speech therapist there has used it to establish several of her students as good candidates for the portable communication devices.

This basic configuration could easily be elaborated to take input from the different input mechanism described previously, and as such would be a total evaluation system, allowing the frontline staff to mix and match, as it were, to achieve the best fit between student and device configuration. These specifications could then be forwarded to this or similar office and a portable device constructed according to the specifications. With the addition of an EPROM programming board, the Apple system can also free the staff from reliance on the computer professional for creation of new vocabularies for the portable devices. Thus the system becomes a highly-evolved development system for the creation and updating of the smaller communication aids.

One more elaboration comes to mind. The portable device could be interfaced to the table top system, granting the portable system users access to the great quantity of educational, word processing, graphic and music generating, and recreational software now available for the Apple computer.

The present system has an automatic testing program, which asks the student to touch word "x". This feature can also be eleaborated to test for other cognitive skills, such as colors, shapes, letters and numbers, word patterns and simple arithmetic forms, multiple choice questions, etc. As such, the system would have definite applications in mainstream, general education, as for the early elementary student.

The application of contemporary electronic technology in the field of communication rehabilitation is one of the great engineering adventures of the decade. Wayne County Intermediate is on the leading edge of that endeavor, which has as its aim the rendering of this technology maximally accessible to the non-speaking individual.

THE DANGER OF THE REDUNDANT MICRO

Donald J. Vinlove

Director of Educational Management Information Systems
Valley View Community Unit School District No. 365-U
Romeoville, Illinois

Microcomputing has experienced explosive growth in the last several years. Marketing has been aggressive in both the hardware and software areas. Education is a natural for the decentralized computing concept offered by the micro. Instructional uses allow individualized programs for students and offer portability to the classroom teacher. Administrative applications begin at the district level and quickly move to the building and department levels. Not only is there danger in "reinventing the wheel" for various applications, but, more importantly, is the real danger of reinventing an existing data base. We must step back and reassess the impact of the micro and develop it in the most advantageous mode.

INTRODUCTION

Since the advent of the microcomputer in 1975, we have been astounded, amazed, awed, and, yes, wary of the promises given by the micro. At first blush, this amazing machine can seem to fulfill our wildest dreams of decentralization, independence, and personal computing power. Upon closer examination, we see certain inherent pitfalls awaiting the unsuspecting user.

We are promised systems in areas ranging from attendance keeping in the classroom to banking. Proud vendors will display hardware and software systems that are seemingly faultless and able to fulfill any purpose or application. The hardware is inexpensive and easy to operate. The software is also inexpensive and able to run within minutes of loading. Documentation offers simple step-by-step directions to arrive at our goals.

The need for personal computers will arise because centralized mainframe data processing is unable to fulfill the needs of turnaround, individual requirements, and interaction with the computer. Another need is decentralization, or distribution, of data processing.

As we listen to and evaluate vendors' "pitches"--as well as staff requests for micro hardware and software--we must consider the matter of redundancy. Care must be taken to ensure that we do not recreate an existing data base already present on an available mainframe. Too many times demographic and biographic data is recreated and not maintained in an accurate manner. A good rule of thumb is: If it's kept somewhere else, don't retain it. . . or recreate it. In other words, avoid redundancy.

REDUNDANCY

There are two types of redundancy to be considered.

1. The recreated data base that exists on a larger mainframe designed to serve the department. This includes demographic and biographic information such as name, number, address, birthdate, sex, grade, school, etc.

2. The recreated or maintained data base kept on like hardware with like purposes, but not shared for the common good. Also included in

this category, is the duplicate data base utilized on incompatible hardware--micro to micro, or micro to mainframe.

I will concentrate on improper, inefficient, or dangerous uses of the micro as opposed to proper or correct uses. A proper use which avoids redundancy would be unique applications not presently available through a mainframe or other micro. Another proper use would be in utilizing mainframe capabilities through downloading of existing information. Still another proper use would be using the micro as a remote or local terminal communicating with a mainframe or network of micros.

FACTORS TO CONSIDER

A negative aspect of the micro is the initial reaction to buy, with little regard for long-range planning. Another negative concern is that of unleashing power in the hands of managers who have not been trained to qualify data. This is quite appropriate to the educational setting in "building" and "department" management--specifically, in the areas of attendance, grade reporting, accounting, inventory, payroll, and purchasing.

Micros are "great equalizers." They can be used to upgrade personnel, customize service, and indoctrinate non-technical people. Great value lies in the changes they can make in staff and students to overcome "computerphobia." The danger lies in creating expectations of great results with little input.

The micro is a business tool. . . not a "magic box." It carries specific risks such as time needed to use systems effectively. Hardware vendors may rush to market various systems that are not completely tested. There is a need for users to work with data processing departments and personnel to best utilize micro hardware. The danger of selecting the wrong software or operating systems can be illustrated by the mention of standards. The CP/M operating system is a standard developed by Digital Resource and is rapidly becoming an industry standard. Selected software must be compatible with compilers available on a particular micro. We can overcome proprietary systems by utilizing standard operating systems and software. American National Standards Institute (ANSI) is working on developing operating standards as well.

USES OF MICROS

In the banking industry, two-thirds of the microcomputers currently in place are less than one year old and cost less than $5,500. Twenty percent of these have data communications, and one-third have letter quality printers. Two-thirds of the people using microcomputers are non-technical and eighty percent are self-taught. These facts can easily carry over to the educational setting by comparing user profiles. Most teachers can be considered non-technical, and many are primarily self-taught in appropriate uses of the hardware. Appropriate workshops are held by various enterprises to ease this gap.

In a recent article in "School Business Affairs," Dennis Costerison reports that of the computers used in Indiana school districts, micros comprise seventy-five percent of those used in academic areas vs. only five percent of those used in administrative areas. Thus, instructional use far outstrips administrative use. Mr. Costerison questions whether micros can handle accounting and bookkeeping tasks of school districts having over 500 students.

One specific area where micros find a valid use is the school library or media center. Library uses include such things as word processing, catalog and card typing, union lists and catalogs of periodicals, bibliography listing, newspaper indexing, general indexing, etc. We must not think "library automation," but rather "service"--service not offered in the past; therefore, a valid use since redundancy does not exist.

PURCHASE PSYCHOLOGY

Underlying psychology of micro purchases has more than a little in common with fad factors involved in the purchase of such things as "hoola hoops" and "frisbees"--or, more appropriately, microwave ovens and cassette recorders. Micros turn up everywhere and are not under the centralized direction of the Information Systems Department. The only control or direction may be by department or building. We must deal constructively with this proliferation. It seems the question is: How are you going to keep them down on the spread sheet once they've seen Visicalc? The prospective user can find a system to satisfy just about any need. The only real problem now is (like a child with too many options on Christmas morning) to pick the system that does the most satisfying job. Qualifying user factors and requirements must be determined before a purchase is made.

FISCAL CONSIDERATIONS

With an increase in reporting and record keeping duties and a decrease in revenues, districts must better utilize resources to survive financial problems of the 80's. Micros have a place here, but we must consider the entire scope and effort. Use of micros will continue to grow as end users see benefits and demand them. Proliferation again needs to be monitored. Final decisions on hardware and software may never be correct, but must be made. We must keep abreast of new software and hardware announcements and evaluate their possible benefits. With proper utilization, the micro can make educational efforts more cost-efficient.

CONCLUSION

As we can readily see, the microcomputer is here to stay. However, its relative newness must be carefully considered in making decisions for its optimum use. A great danger lies in recreating existing data bases, or creating a base of unmanageable proportions. Great amounts of resources can be wasted on a redundant effort. The emerging micro is the challenge of the 80's to be met in our personalized business and instructional world of education.

REFERENCES

School Business Affairs--January 1982, page 14, "School Business Is Going Micro"

Chicago Tribune--February 2, 1982 (UPI) "Experts Promote Software Compatability"

ICP Software Business Review--Spring 1982, "Personal Computers: Not Just for Whiz Kids Anymore"

Information Systems Planning Service--January 1982, "Industry Preview" Research Memorandum

Small Systems World--February 1982, page 28, "MICROS tied to MINI to Test Remote Data Entry"

MIS WEEK--February 3, 1982, "Small Banks Told: MICRO AN EQUALIZER"

Computer World--February 2, 1982, page 134, "MICROS & Data Processing Management"

Library Applications of Microcomputers
By: Hines, Winkel, Collins, Harvey
University of North Carolina
at Greensboro

Reporter, New York State Association of School Business Officials
January 1982, page 8, "NYS Bauds"

IMPLEMENTING INSTRUCTIONAL COMPUTING AND COMPUTER LITERACY IN A SCHOOL OR COLLEGE

Frederick H. Bell
University of Pittsburgh

ABSTRACT

This paper contains guidelines for setting up instructional computing and computer literacy programs in schools and colleges. Consideration will be given to typical problems. Decision making strategies with respect to planning, modes of computer use, hardware selection, hidden costs, conceptual and physical location of computers, and faculty development will be discussed.

INTRODUCTION

The process of implementing instructional computing and computer literacy programs in a school system or college is quite different from developing programs in other areas. First, instructional computing and computer literacy are cross disciplinary; consequently many people must be involved at the beginning. Second, strong ties must be established among data processing personel, administrators, and classroom teachers. Third, significant numbers of faculty members must be educated in instructional computing and computer literacy. Fourth, it is critical to avoid piecemeal planning. While individual initiative of an administrator or a faculty member (sometimes due to the availability of outside funding) may provide the impetus for these programs, their ultimate success depends upon broad faculty and administrative participation.

PLANNING AND ORGANIZATION

At the beginning of the planning process, three types of people are needed: people who have done it before, people with answers to the six concerns in the ABSTRACT above, and people who will be implementing the program. The planning group should include someone from a similar educational institution who has had a major part in setting up an instructional computing and computer literacy program, someone who knows computer hardware and software, a person who knows the ins and outs of courseware, administrators, faculty members, and someone from the physical plant.

Contrary to what is typical in this relatively new field, implementation in any of the five non-planning areas should not be started until planning is complete in all areas. With careful planning, most of the problems inherent in the procedure of "buy it, find a place to put it, figure out what to do with it, then find people to do it" can be avoided. If an institution does not have appropriate personnel for planning and implementation in all areas, it will be necessary to locate outside people or consider postponing development until appropriate personnel are available.

After the planning group has set policies, it should continue monitoring and supporting the people involved in all phases of implementation. Sometimes a proposal for outside funding of instructional computing is written by an individual, quickly, in time to meet a deadline. When funding is forthcoming a project manager is hired to implement a sketchy plan. Subsequently, the job of the manager is to redeploy funds or obtain new funds to attack problem after problem that arises due to inadequate planning. In such cases money is wasted and the system-wide impact of the program is less than hoped for.

WAYS TO USE COMPUTER HARDWARE

It should be noted that instructional computing and computer literacy need not be separate programs. In fact, usually they should be carried out as a unit. Good instructional uses of computers promote computer literacy for teachers and students who are using computers in their classes. Conversely, computer literacy courses in schools and colleges can incorporate using computers to enhance teaching and learning. Consequently, the two will be discussed as a unit.

Among modes of instructional computing are record keeping, classroom management, learning new concepts, practicing skills, learning decision making and problem solving, and developing independent learning skills. Today computers are used in education in most subject areas, from teaching typing to simulating laboratory experiments.

As the name implies, the first computers were designed to be accurate, high speed arithmetic machines. However, modern computers are general information processors that can handle text, graphics, color, and logic. The variety of subjects in which computers are used and uses within each subject are limited, primarily, by the availability of hardware, software, courseware, and creative instructors.

In general, computers in school and college programs are used for faculty development, to enhance instruction, to facilitate learning in various subjects, and to develop computer literate students. For faculty members, instructional computer literacy includes knowing how to operate computers and being able to select, evaluate, modify, use, and even write courseware for student use. For students, classroom computer literacy includes learning how to load instructional programs into computers and how to interact with these programs. At various times in school and college, it is necessary for instructors to teach students study and test taking skills. When students use courseware, they must be instructed in how to boot up microcomputers, load programs from disk or tape, respond to computer displayed instructions, and even how to use the keyboard.

General computer literacy encompasses areas such as the history of computers, the nature of computers, communicating with computers, types of computers and peripheral hardware, capabilities and limitations of computers, computer uses, and programming skills. Computer awareness may only require using computer programs, but computer literacy involves experience and proficiency in programming. Literacy levels range from knowing the children's language LOGO to facility with BASIC to knowing PASCAL or FORTRAN.

SELECTING A MICROCOMPUTER

The key question in selecting computer hardware is NOT which computer is best; it is what will the computer be used for. For example, both the TRS-80 Model II and the APPLE 2 are "good" computers. However, the TRS-80 Model II is better suited for business applications courses and the Apple for children's use in elementary schools. Currently many school systems and colleges (some of which have time sharing computers) are buying microcomputers for instructional computer literacy. Following are some answers given by decision makers in schools and colleges to the question: "Why did you select 'brand X' microcomputers?" for use in your institution

*For every two we bought, we got one free.
*A friend likes his.
*I've used that one before.
*It has great color graphics.
*It's what other schools bought.
*They are reliable machines.
*Good software is available.
*Brands X and Y are no good.
*We didn't like brand X's hiring policies.

All these reasons have merit; however the people who got one free for buying two, ended up with business oriented computers (that had to be shipped out of state for service) for use in a junior high computer awareness program. The people who bought 12 brand Xs for the great color graphics, got black and white monitors to save money.

Every year articles on selecting microcomputers appear in several of the journals and magazines in the bibliography at the end of this paper. Since a new and improved version of most microcomputers comes out every few years, each purchase decision must be made with respect to the "state of the technology." Microcomputer evaluation and comparison articles, together with general "how to select hardware" articles are a good place to start. For example, see the Fall 1979 AEDS Journal special issue on Microcomputers. Instead of comparing today's microcomputers or writing another set of generic guidelines for selecting computers, the following discussion will examine some of the common mistakes that people make in their selections.

Mistake #1 (Buy It Where It's cheapest): While cost is certainly a factor, vendor assistance in setting up hardware and helping to train faculty, as well as fast, reliable service, are important factors in making a selection. When you can't figure out why something won't work, is an answer available by phone from the dealer? If you buy a printer, disk drive, computer, monitor, and word processor from five different dealers, you will be on your own in determining how to hook up the various components and in diagnosing problems. Unless in house repairs and service are available, beware of dealing with too many vendors.

Mistake #2 (Get All The Options): Color graphics and sound are impressive, but are they really needed? When color, sound, or animation are integral parts of the courses in which the computers will be used, buy machines that do them well. However, these frills can be distractors when not integral to the learning task and their novelty soon fades for many students.

Mistake #3 (Buy Bottom of The Line Systems): Four K (about 4,000 characters) of user memory is not enough to do instructional computing. And it's barely adequate for beginning computer literacy. Even 16K machines are inadequate for a lot of the good educational courseware. For many educational uses, 32K of user memory is needed. If you do go with the $399 or $599 micros, be sure to find out about the availability of various additions and their cost, in case you should need them later.

Mistake #4 (Too Much of The Same) Before buying, for example, 12 brand X computers, consider your diverse needs. It may be that most of the time students and faculty in a computer literacy program can get by with 16K, black and white monitor, cassette tape computers. However some faculty and students may need to do large programs, word processing, or color graphics. It may be better to buy 10 systems; two of which are equipped with 48K of user memory, one or two disk drives, and color monitors for use in the exceptional, but necessary, cases.

Conversely, a teacher education college planning to offer general computer literacy and instructional computing courses for students, who need to be familiar with a variety of microcomputers, should diversify. In this situation two each of middle of the line Apple, TRS-80, Atari, Texas Instruments, and Commodore computers may be the appropriate decision.

In summary, your school or college microcomputer needs are unique to your own situation. This means that good planning should begin with analyzing the capabilities and limitations of several widely used (in education) microcomputers that are well supported by software and courseware vendors. Primary information sources to determine machine capabilities are magazine articles, and professional organizations, together with hands on demonstrations and short seminars by the various vendors. Then microcomputer selection should be determined by a precise evaluation of how the computers will be used, for what purpose, and by whom. Final selection will require the combined expertise of classroom instructors who will be using the computers and by local people who know hardware and software. Some key questions to be answered are:

1.Does it have a standard keyboard?

2.Is the screen easy to read?

3.How many graphics and text characters per screen line and how many lines?

4.How are the graphics?

5.What are its disk and internal memory capabilities?

6.Does it have sound and color?

7.Can it do word processing?

8.What are the add ons and what do they cost?

9 Does it have standard BASIC?

10.What special BASIC functions does it have?

11.Are other programming languages available?

12.Is courseware available?

13.How portable is it?

14.What start up support is available?

15.How and where is service available?

16.How much is a service contract?

17.What are the warranty terms?

18.Are cables, connectors, etc. included?

19.What is the quality of the manuals?

20.When will it be delivered?

21.Who can be called for help and advice?

22.What is the educational discount?

23.What are its special strengths?

24.What are its quirks?

25.Can the microcomputers be networked?

26.Can you talk to someone who is using these machines?

27.Is this really the microcomputer for your needs?

Many schools and colleges that are considering purchasing microcomputers are presently using time sharing computer systems for instruction and computer literacy. For example, the science and mathematics department may be using time sharing for simulations and tutorials and FORTRAN or COBOL may be taught on the system. Effective and efficient use of resources may dictate using a combination of time sharing and microcomputers. For instance, a school district in Pennsylvania is using Apple microcomputers for instructional computing and computer awareness in grades 1 through 6, and Apples and time sharing for computer literacy and BASIC programming in grades 7 through 9. The district's mainframe system is used to teach COBOL, FORTRAN, and BASIC in grades 10 through 12.

HIDDEN COSTS

Among the hidden costs of maintaining a stable of microcomputers are service, supplies, courseware, and upgrading. Service contracts can cost as much as 35% per year of the original purchase price of the hardware. If the computers are not going to be in heavy use and are used properly, pay as you go service may be cheaper than a contract.

If several printers are connected to heavily used microcomputers, paper and ribbon costs will begin to add up. It is easy to overlook items such as dust covers, printer stands, tables, extension cords, switch boxes, carrels, diskettes, and tapes. Some electrical work may be needed to set up a computer lab. Security is another cost. Special door locks and bolts as well as security pads for the computers may be needed. Personnel to oversee the lab and assist faculty and students may also be required. Courseware is another expense. Courseware disks can cost from $20 up to $100s. A printer is virtually a necessity. And of course disk drives are required for serious work. Color monitors, if required, are also costly. Then it is well to budget for additional computers and expansion. Microcomputers do wear out and heavy use may dictate enlarging the computer stable. New applications, new generations of computers and new software may make additional upgrading desirable.

Faculty development and training is also expensive. In many schools and colleges it happens that after purchasing and equipping a lab, few faculty members are using it for instructional computing. So faculty development is necessary, which may mean hiring a consultant ($2,000) to give seminars after classes for 20 or 30 faculty members at $300 per person. Yes, it may be necessary to pay faculty for training sessions. Then there are magazine subscriptions and other miscellaneous costs.

For example, in my own experience, to set up and support a lab housing $20,000 worth of microcomputer hardware, it cost $10,000 for security hardware, electrical equipment, and minimal furnishings. Additional personnel costs run about $10,000 per year. Supplies, upgrading hardware, service, and courseware add another $5,000 per year. These figures do not include space and overhead costs. That is to say, to maintain, support, and upgrade a microcomputer lab could cost as much per year as the initial price of the hardware. And, in this example, we're gambling that we will save 50% on service by not having a contract.

CONCEPTUAL AND ACTUAL LOCATION

Physically, a good location for microcomputers is a classroom or laboratory reserved exclusively for computer activities. The lab should be in a secure area, out of traffic, with adequate supervision. The main security problem is theft, not vandalism. For approximately $100 per unit, computers can be secured to tables with locked pads that can be unlocked to remove them for service or temporary relocation. Appropriate supervision may be a secretary or student assistants.

Conceptually, instructional computing and computer literacy should be institution wide. That is, computer literacy is becoming a basic skill for school and college students. Instructional computing should be available, as hardware permits, to instructors and students in all subject fields. It usually does not work well to make computer literacy or instructional computing a subsidiary of one department. Computer literacy courses will naturally be offered by departments having personnel qualified and willing to develop and teach them.

FACULTY DEVELOPMENT

Faculty development may be as costly as hardware and it is difficult to carry out. To set the stage for faculty development, consider the impetus for developing instructional computer literacy programs.

In most situations the impetus comes from a few instructors, or an administrator, with appropriate support and pressure from community and students. Schools and colleges most likely to go into instructional computing and computer literacy, using microcomputers, are those that already have been doing it, to some extent, with time sharing. The flexibility, portability, and visibility of microcomputers make them well suited for various classroom applications.

It seems that instructors and administrators who are instrumental in obtaining microcomputers for a school or college are, themselves, computer literate. They may also have been involved in instructional computing. Sometimes these people fail to realize that non-involved personnel choose to be non-involved. Just buying microcomputers won't, alone, get most of the faculty and administrators involved. Their involvement will come through incentives and subtle pressures.

An effective way to get uninitiated school and college teachers on the microcomputers is to pay them to attend training seminars, with required homework. Such seminars should be taught by someone who knows computers, computing, teaching and learning, and curriculum development. This person should have at least some modest experience in courseware development and his or her main concern should not be programming.

Training sessions should include a blend of learning to operate hardware; viewing, using, and evaluating courseware; learning basic BASIC; and writing some instructional programs. Experience indicates that for 10 trainees, 1 will be there solely for the pay; 2 will have difficulty mastering hardware and algorithmic thinking skills; 5 will do well, but will not pursue it later; and 2 will do instructional computing in their classes.

CONCLUSION

In conclusion, there is much more to implementing instructional computer literacy than buying hardware and courseware. The key to a successful program is appropriate planning with sound goals and objectives for implementation, together with enlightened faculty involvement.

BIBLIOGRAPHY

AEDS 1981 CONFERENCE PROCEEDINGS, see articles on pages 12, 119, 229, 234, 243, and 270

AEDS Journal. Special Issue On Microcomputers, Fall 1979.

Billings, K. and Gass, S. "Adding a Microcomputer to Your School Picture." ELECTRONIC LEARNING, Jan/Feb 1982.

Finkel, L. "Buying a Micro: What Every Educator Should Know." ELECTRONIC LEARNING, Jan/Feb and Mar/Apr 1982.

Hedges, W. "How to Introduce Teachers, Principals and Curriculum Personnel to the Microcomputer." THE COMPUTING TEACHER, Vol 8, No 5.

Miastkowski, S. "Choosing Your Popular Computer."POPULAR COMPUTING, Nov 1981.

MicroSIFT NEWS. Oct 1981 issue on selecting microcomputers and evaluating courseware. Northwest Regional Educational Lab, Portland, Oregon.

POPULAR COMPUTING. "Battle of the Best." Four lead articles on brand name microcomputers, Jan 1982

Press, L. "Getting Started in Personal Computing." OnCOMPUTING, Spring 1981.

Title Page

Addressing the Need for Pre-College Teacher Training in Computer Science in the State of Georgia.

Drs. Frank C. Clark and
Harrison S. Carter
Georgia Southern College
Box 8093
Statesboro, GA 30460
(912) 681-5286, 681-5390

Correspondence should be with Dr. Clark

In Georgia as in most of the states (all but about four) there is no certification program for teachers in the area of computer science. Very few teachers are teaching computer-related courses in the high schools for several reasons not the least of which is because they have received little or no training in this area. Lack of accessibility to hardward of course is also a major detriment.

During the summer of 1979, the authors sent questionnaires to some 550 high school mathematics teachers in the state of Georgia. Of the 550 teachers contacted some 270 responded. From the results it was determined that the average number of computer related courses taken during their college training was one course. This pointed our the woeful lack of training in the computer science area. The survey also indicated that the respondents sensed a pressing need for implementing computer related courses in the high school curriculum. How could this be accomplished if the teachers are ill prepared?

In an attempt to ameliorate in a small way this problem, the authors submitted a Pre-College Teacher Development proposal to the National Science Foundation for a four-week Summer Institute to be held at Georgia Southern College. This proposal was accepted and the institute was funded for the Summer of 1980.

The objectives of the institute were:
(a) Introduce and strengthen knowledge of high school teachers in the use of computer concepts.
(b) Assist the pre-college teacher in integrating computer concepts into current curriculum.
(c) Provide assistance and guidance in obtaining computer capability.
(d) Establish a basis of contact in order to provide "on the job" assistance, as well as a forum for discussion and exchange of ideas on instruction

Characterictics of Participants

The institute was designed for 30 participants. Participants were selected from those applying who indicated the greatest need for, interest in, and potential use of this type of program. The need was established by considering the amount of previous course work and/or experience in computing. Additionally, the potential for the applicant to transfer some of this information to other teachers in need was considered. Finally, the potential for the applicant to successfully put this increased knowledge to work in the high school setting was considered.

Format and Instructional Mode to be Used

The instructional format used involved regularly scheduled lecture/discussion periods. Additionally, demonstrations and laboratory periods were used to supplement the lectures/discussions. A typical daily schedule was as follows:

9:00-10:15	Lecture/Discussion Period I
10:15-10:45	Break/Informal Discussion
10:45-12:00	Lecture/Discussion Period II
12:00- 1:00	Lunch-Office Open for Consultation
1:00- 1:50	Lecture/Discussion Period III
1:50- 2:10	Break/Informal Discussion
2:50- 3:00	Lecture/Discussion Period IV

3:00-5:00	Supervised Laboratory/ Demonstrations
7:00-9:00	Unsupervised Laboratory - Terminals Available. Tutoring as requested.

This schedule called for 250 minutes of instructions per day for twenty days. This resulted in 5000 total minutes of formal instruction during the four-week institute. This is consistent with the University System of Georgia Requiring 2500 minutes for 5 credit hours. Since the participants earned 10 hours of credit, they received 5000 minutes of instruction.

The laboratory periods were supplemental but essential to round out the total program and learning experience. The thirty participants were divided into two groups for the lecture/discussion periods. This greatly enhanced the learning experience and enabled the instructors to analyze the individuals' needs and situations of the participants regarding the practical alternatives they had for obtaining computer support. This was a key and unique feature of this institute. Not only did we teach the fundamentals of computing and computer programming, but we also helped each participant to develop a plan for obtaining computing capabilities at his/her school. Additionally, we discussed and developed strategies for integrating computing into the curriculums of the schools of the participants attending the institute.

Although the schedule appears quite structured, all of the instructors were very flexible and sensitive to the needs of students. Therefore, when the need arose, we did not hesitate to modify the schedule to enhance the effectiveness of the institute.

The allocation of time per topic is described below. Basically the institute consisted of course content modified for this particular purpose and was taken from two courses: (1) Computing Concepts and (2) BASIC Programming.

Computing Concepts

Topic	Allotted Time
Introductory History and Rationale	1.0 Hrs.
What is Computing	1.0
Computers in Action - Applications	2.0
Overview of Computer Software	1.5
Principles of Programming	1.5
Overview of Computer Hardware	1.5
Data Representation	3.0
Input/Output Devices	2.0
Secondary Storage Devices	2.0
Communications	2.0
The Central Processing Unit	2.0
Hardware Systems	3.0
Survey of Programming Languages	1.5
Evaluation and Selection of Hardware	3.0
Evaluation and Selection of Software	3.0
Alternatives for Computing in High Schools	3.0
Integrating Computing into the High School Curriculum	3.0
Implementing a Plan for Computing at Your High School	3.0
Computing in the Future - What's Ahead	1.0
Testing of Participants	2.0
Total Hours	42.0

BASIC Programming

Topic	Allotted Time
Programming at Georgia Southern College	1.0 Hrs.
Flowcharting	1.0
Decision Tables	1.0
System Commands and Interactive Programming	2.0
Programming Using Batch Processing	1.0
Using Already Written Programs	2.0
Unformatted Input/Output	1.0
Assignment Statements and Variables	2.0

Unconditional Branching	.5
Conditional Branching	1.5
Loops	2.0
Arrays and Subscripted Variables	3.0
Functions	2.0
String Operations	1.0
Matrix Operations	2.0
Data Files	3.0
Formatted Input/Output	4.0
Application Programming	10.0
Testing of Participants	2.0
Total Hours	42.0

How the Overall Project Operating Plan Was Designed to Achieve Project Objectives

Basically the operating plan for the project was designed to meet the objectives of the institute by employing the instructional mode described earlier to teach/discuss computing concepts, computer programming, practical computing alternatives, current computer hardware including costs, and implementation/integration or computing in high school curriculums. It was felt that these topics were essential to prepare the participant to implement a high school program utilizing computers.

In order to achieve the first and salient objective, the institute was designed so that a participant was exposed to about 4 hours of lecture/demonstration/discussion plus 2 hours of "hands-on" time at the computer terminals per day. At the onset of the institute, it was determined how much, if any, knowledge of computers the student had. Each student was encouraged to participate in the discussion sessions and ask questions during lecture and demonstration. Also individual tutoring was held so that each participant could ask questions about particular concepts.

The second objective of integrating computer concepts into current high school curriculums was accomplished primarily by consultation of the participant with Drs. Clark and Carter since both had taught high school students in the computer related fields. The text that was used for the institute was one that could be used by the teachers for their high school instruction. It was felt that this would assist in meeting this particular objective since the content and organization of the textbook had some influence on the structure of the course.

The project staff discussed ways and means whereby the participants could obtain computer capability for their schools. All three staff members were knowledgeable about the state-of-the-art in computing equipment, prices, and manufacturers. Staff members were in contact with personnel at the central site in Athens and provided participants with names and addresses of key individuals. Staff gave advice as to what type of equipment would be best in a given situation.

The staff made visits to the various high schools to provide "on the job" assistance and consultation. The staff served as a forum for discussion and exchange of ideas on instruction. Also, telephone conferences were held to solve problems and maintain contact.

PARTICIPANT SELECTION PROCEDURE

Contact had already been made with the State Department of Education - Division of Mathematics and Science Education, the Georgia Science Teachers Association and the Georgia Mathematics Teachers Association. These three organizations agreed to provide addresses of all potential participants in the state. Additionally, they agreed to help publicize the institute.

A descriptive brochure, cover letter and questionnaire was sent to all identified potential participants. The questionnaire aimed at determining the current knowledge of computing, need for the program, interest in the program, and potential use of the material contained in the program. This data allowed for an initial screening of applicants. Final selection also involved a statement from the applicant's Principal. Finally, preference was given to applicants in the immediate service area of Georgia Southern College and to others who had no viable alternative opportunities.

As we mentioned earlier a survey of some 550 Georgia high school mathematics teachers was conducted during the summer of 1979. Of the 550 surveyed some 260 responded (47.2%).

The results of the survey are listed below:

(1) Enrollment of the school in which you teach.

Size	Frequency of response
<500	21
500-1000	103
1000-2000	104
>2000	32

(2) Populations of community in which your school is located.

Size	Frequency of response
<10,000	58
10,000-30,000	90
30,000-100,000	51
>100,000	53

(3) Are any computer facilities available to your school?

Response	Frequency of response
yes	130 (51%)
no	126 (49%)

(4) Are computers currently being used in your school for instructional support in mathematics?

Response	Frequency of response
yes	87 (34%)
no	172 (66%)

(5) Do you feel that there is a need for teaching computer courses in high school?

Response	Frequency of response
yes	87 (34%)
no	172 (66%)

(6) Do you feel that the computer area should be an area for state certification?

Response	Frequency of response
yes	149 (59%)
no	102 (41%)

(7) Would computers be useful in the teaching of high school mathematics?

Response	Frequency of response
yes	242 (93%)
no	18 (7%)

(8) The average number of graduate or undergraduate courses in data processing, computer science, computer programming or other computer oriented area for those responding was 1.25.

(9) Georgia Southern College hopes to be able to offer a four-week tuition-free institute consisting of 10 credit hours of graduate work in the computer field.

The institute will focus on the following areas:

(a) Computer concepts
(b) Computer programming
(c) Use of computers in secondary mathematics
(d) Sources of computer capabilities for public schools

Would you attend such an institute?

Response	Frequency of response
yes	193 (78%)
no	55 (22%)

EFFECTS OF PROGRAMMING ON MATHEMATICS ACHIEVEMENT

Mary C. Harrison

Graduate student, Department of Mathematics and Statistics, University of Central Florida

INTRODUCTION

In 1972, Paul G. Watson wrote "...the question of whether the computer should be used in education and/or instruction is now no longer relevant...the relevant question today is ...how they will affect the school's instructional processes in the next decade..."[1]

Now it is the next decade. And we are still asking the same question. How can a computer be used most effectively in the classroom?

Is it sufficient for students to have a passing acquaintence with computers (computer literacy)?[2,3,4] Should everyone develop the skills of a programmer?[5,6] (Or is this really asking the same question?)[7,8] Will computers fulfill their claims of greater student achievement and motivation?[9,10,11]

While it is easier to ask questions than to provide answers, the need for answers is acute. The number of computers in the classroom is on the rise.[12,13,14,15] Computers are here to stay.

So far, the emphasis on computers in education has been upon the student as a passive recipient of knowledge. Thus most research has been dedicated to determining how computer-aided instruction (CAI) affects a student's understanding of concepts.[16] While this is important, it is not enough.[17][18] A computer is more than an electronic workbook; it is a tool. Like a pencil, it can be manipulated by the student to explore ideas and make conclusions.

This idea of active student participation is not new. The University of Pittsburgh, supported by the National Science Foundation, initiated Project Solo (called Soloworks) in 1969.[19] Courses in "computer science" and "computer mathematics" are a part of the curriculum in many high schools. Universities offer mathematics classes with a computer option.

But how do students in these classes fare? Do students who "teach" the computer via algorithms develop a measurably better understanding of the concepts involved? Do their skills increase? Do they develop better problem solving skills?

Research in this area is scant and the results conflicting. There have been very few studies whose aim was to detect the effects of programming on achievement.

This experiment was designed to consider three such issues:

(1) How do programming students' mathematics achievement compare with that of non-programming students?

(2) Do programming students develop better computational skills than non-programming students?

(3) Do programming students develop better problem solving skills than non-programming students?

It was thought that these ideas could be examined most easily by controlling the composition of the student's homework assignment rather than affecting the teacher's presentation of the topic.

PRELIMINARY WORK

Two pilot studies were run before the main experiment was attempted. The first one provided guidance in the composition and length of the homework assignments. The second one tested a series of three computer-aided instruction (CAI) lessons written by the author. The topic for each study was taken from elementary trigonometry--The Law of Sines.

Pilot one:

This study was performed at Easton High School, Easton, Maryland and was divided into two distinct parts. The first part measured the time a student spent writing computer programs while the second part measured the time he spent on drill and word problems.

A total of two teachers and fifty-seven students participated in the study. The students were seniors, juniors, and a sophomore who were taking a college preparatory mathematics class and/or a computer mathematics class. Nine of these students participated in the first part and fifty-six were involved in the second.

Part one:

The nine students who participated in this first part of the study were given the opportunity to work as many as four computer programs dealing with the Law of Sines. Some students worked all four of the programs while others did only two.

The students were allowed only one problem at a time and could not receive another until the previous problem had been completed to their satisfaction. The programs were assigned in a random manner. Each program slip had a space for the student's name, a space to record his starting time, the statement of the problem, and a space to record his ending time. Two of the programs involved the ambiguous case.

Part two:

The students who participated in this part of the study were given an in-class two page test on the Law of Sines. Four forms of this test, each containing five problems, were used. The test forms were randomly distributed among the students.

The first page of each test had a space for the student's name, a space to record his starting time, four drill problems, and a space to record the time when these four problems were completed (intermediate time). The second page contained a space for the student's name, one word problem and a space to record the student's final time. Half of the problems required the use of the ambiguous case.

RESULTS AND CONCLUSIONS

Using an analysis of variance procedure (ANOVA), it was determined that the mean times spent working on a computer program, drill problem or word problem did not vary according to the problem chosen. Also, the mean times did not vary between teachers.

Thus it was feasible to determine an overall mean time for each type of assignment. To determine which means were different, Duncan's multiple comparisons test was run. It showed that differences existed between the mean programming time and the mean times for working drill problems and word problems. There was no statistical difference between the mean time for working a drill problem and the mean time for working a word problem. (see table 1)

TABLE 1

DUNCAN'S MULTIPLE RANGE TEST
(Time in minutes)
(Means with the same letter are not
significantly different)

ALPHA LEVEL=.05 DF=112 MS=127.125

GROUPING	MEAN	N	TYPE
A	30.375	24	3
B	8.512	43	2
B	6.984	48	1

Although the difference between the mean times for solving a word problem and working a drill problem did not test to be statistically significant, the author wished to be conservative and used the mean times as determined.

Pilot two:

In all previous research, there has been no attempt to compensate for a possible Hawthorne effect resulting from the use of a computer. Thus the author developed a series of CAI lessons to be used in the main experiment. Each lesson reviewed a different facet of the Law of Sines: its derivation, the non-ambiguous case, and the ambiguous case. The lessons were designed to run on a Radio Shack Model I or Model III computer with 16 K of memory.

These lessons underwent three stages of testing. They were first examined by the author's husband, John E. Harrison. He reviewed them for accuracy and bullet-proofing (ways to keep a program from breaking down). The next review was by the author's advisor, Dr. Lee H. Armstrong. He examined them for pedagogical content.

The final stage of testing was

done with eight students from Dr. Armstrong's college trigonometry class. These college freshmen had just finished studying the Law of Sines. They reviewed the lessons for clarity of presentation. The students were timed to provide an indication of each lesson's length. Mean times were calculated for each lesson (see table 2).

TABLE 2

CAI MEAN LESSON TIMES
(Time in minutes)

	N	MEAN
Lesson 1	2	10.0
Lesson II	5	14.4
Lesson III	3	26.3

THE EXPERIMENT

Once this background work was completed, it was time to run the main experiment in one of the local high schools. Fourteen students from the West Orange High School elementary trigonometry classes were chosen to participate in the six day experiment. All of these students were currently taking or had taken a computer programming course at the school.

The students were divided into three groups based upon their previous semester's grade in trigonometry. Each group contained students who had received high grades and students who had received low grades. All three groups were taught the Law of Sines (including the ambiguous case) by their regular teacher. Only their homework assignments differed. Using the results of the pilot studies, the composition of each group's assignments were adjusted so they were approximately equal in length (timewise).

The control group (group C) had standard, paper and pencil, hand-written assignments consisting of drill and word problems. There were six students in this group.

The computer programmers (group A) were assigned very few paper and pencil exercises. Rather, they were given a page with four programming problems concerning the Law of Sines. Each student chose three programs to write and run. There were four students in this group.

The computer users (group B) worked some, but not all, of the problems given to the control group. In place of the other problems, the students in this group reviewed the day's lesson with the CAI materials written by the author.

Within the CAI lessons were imbedded the computer programs assigned to the experimental group (group A). Thus it would be possible to determine if the use of canned programs is as effective as the writing of the programs. There were four students in group B.

In addition, all students kept an activity log where all time spent on each type of assignment was recorded.

At the end of the experiment, all students took the same forty point, five question test. The test contained no programming questions. The first three problems tested for mechanical skills and were worth a total of eighteen points. The last two questions were word problems worth a total of twenty-two points.

RESULTS

Because test scores tend to be normally distributed, the results were first examined by an analysis of covariance procedure where the covariate was the total time spent on the homework assignment. The test results were considered in three ways:

(1) total test score
(2) skill problem score
(3) word problem score

Although there was a difference in the means, none of the differences were statistically significant at a .05 alpha level (see table 3).

Due to the large variability in the scores and the small sample size, a non-parametric test was applied (the Kruskall-Wallis H test). As this procedure does not require the assumption of normality, the effects of "wild points" are minimized.

However, this procedure also showed a lack of significance at the .05 alpha level (see table 4).

TABLE 3

ANALYSIS OF COVARIANCE RESULTS

	Group A		Group B		Group C			
	mean	std. dev.	mean	std. dev.	mean	std. dev.	F-value	p-value
Total Score	22.5	12.23	26.3	7.93	19.8	8.13	1.27	.3314
Drill Score	12.8	6.40	13.3	3.20	10.7	4.37	1.28	.3302
Word Problem Score	9.8	7.50	13.0	7.39	9.2	5.64	.53	.6102

TABLE 4

KRUSKALL-WALLIS H TEST
(Rank sums for each group)

	A	B	C	H-value
Total Score	29	38	38	1.39
Drill Score	32.5	34.5	38	.8452
Word Problem Score	25	36	44	.8810

CONCLUSION

Despite the lack of significance in these tests, it can be noted that:

(1) Programming students had at least as good an understanding of the Law of Sines as the control group.

(2) The replacement of paper and pencil exercises with computer exercises did not detract from the students' ability to perform the calculations required for the Law of Sines.

However, it should again be noted that the experiment dealt with a very small sample size and a limited time span. Perhaps this had an effect upon the results. It would be interesting to repeat this experiment over a semester course with larger numbers of students in each group. This would allow more homework problems and computer programs to be assigned. More tests could be given. And thus a more complete evaluation could result. However this would also require the development of more CAI materials.

This study does raise a question concerning the role of CAI in education. Although none of the means were statistically different, they were highest for the CAI group. Should computer-aided instruction materials be used as an adjunct to rather than a replacement for regular classroom instruction? Additional study could be conducted to determine which of the following methods results in optimal effectiveness of CAI materials:

(a) The student would work a prescribed number of paper and pencil exercises at the end of a lesson.

(b) The student would review a topic via a CAI lesson and then work a reduced number of paper and pencil exercises.

(c) The student would rely totally on the CAI materials for instruction and homework.

Perhaps such a study could be combined with a more extensive examination of the effects programming has upon achievement. The results from such work may help to define the proper role of computers in the classroom.

REFERENCES

[1] Paul G. Watson, "Using the Computer in Education. A Briefing for School Decision Makers," Educational Technology Publications, 1972, pp.3-4.

[2] Charles J. Zoet, "Computers in Mathematics Education," Mathematics Teacher, November 1969, pp. 565-566.

[3] Ludwig Braun, "Computers in Learning Environments--An Imperative for the 1980s," Stony Brook, 1980, p. 6. (Mimeographed.)

[4] Ronald E. Anderson, Daniel L. Klassen, and David C. Johnson, "In Defense of a Comprehensive View of Computer Literacy--A Reply to Louhrmann," Mathematics Teacher, December 1981, p. 688.

[5] W. Hoffman, "Computers for School Mathematics," Mathematics Teacher, May 1965, p. 325.

[6] Robert [illegible]. Gravina, "Computer Program Analysis," Computers and Education, 3 (March 1979): p. 195.

[7] Jane D. Grawronski, "Computer Literacy and School Mathematics," Mathematics Teacher, November 1981, p. 614.

[8] Arthur Luehrmann, "Computer Literacy--What Should It Be?," Mathematics Teacher, December 1981, p. 683.

[9] Zoet, p. 564.

[10] Joan Danver, "The Computer as a Study Tool," National Association of Secondary School Principal Bulletin, February 1970, p. 19.

[11] Lisa Schloss and Leslie D. Ball, "Computerized Education: Should We or Shouldn't We?," AEDS Monitor, July/August/September 1981, p. 20.

[12] Braun, p. 6.

[13] Christine Copple, "Computers in the Secondary Mathematics Curriculum," Stony Brook, January 1981, p. 22.

[14] Richard A. Park, Geoffrey Goldbogen and Linnda R. Caporeal, "RPI gives Ataris to freshman to study effects on campus," Electronic Education, January 1982, p. 7.

[15] Marjorie Blair, "Get out your computers," Electronic Education, November 1981, p. 11.

[16] Kathleen Clower, "Instroducing technology's as simple as boiling water," Electronic Education, November 1981, p. 22.

[17] Frederick H. Bell, "Can Computers Really Improve School Mathematics?," Mathematics Teacher, May 1978, p. 433.

[18] LeAnne Brown, "Its theme: 'I do and I understand'," Electronic Education, December 1981, p. 19.

[19] Thomas A. Dwyer, "Soloworks: Computer-Based Laboratories for High School Mathematics," School Science and Mathematics, January 1975, pp. 2-9.

COMPUTER SIMULATION LABORATORY FOR DESIGN OF EXPERIMENTS

R. A. Hirsch, J. O. Geremia, C. Wu

Mechanical Engineering Department
United States Naval Academy

ABSTRACT

In any course concerned with the design of experiments students are introduced to the theory and calculations associated with various experimental plans such as factorial or Box-Behnken designs. In order to implement these plans realistic experimental data in needed, but in the early use of these plans emphasis should be on the mechanics of their utilization and not on the difficulties of obtaining real data. For this reason a computer simulation has been developed wherein the student can select values of a number of independent variables and the computer responds with the values of the dependent varibles including random errors. The student can then conduct "experiments" according to one of several plans in order to discover the main and interaction effects, the magnitude of the overall "experimental" error, and the nature of the functional relationship among the variables. The simulation program and its application are described.

INTRODUCTION

In a required laboratory course at the Naval Academy, senior mechanical engineering students are presented with the following hypothetical situation: Experiments are to be conducted on a "black box" representing a motor, machine, or process. Three independent variables are controllable within specified ranges by the experimenter. Two dependent variables are to be measured and random errors of unknown magnitude will occur. The main objective is to find one or more sets of operating conditions which will maximize one dependent variable and minimize the other. The secondary objectives are to discover what sort of functional relationship exists among the variables and the magnitude of the random errors. The hypothetical experiment can be posed in any concrete terms the instructor chooses.

The "experiments" are conducted utilizing a computer simulation to be described and initially the students are free to experiment in any way they choose. Most students employ the "one variable at a time" strategy and after 30 or 40 runs have discovered at least one "answer."

The factorial design is then described and the students then rerun the "experiment" utilizing the 3 factor, 2 level design and with 20 runs are shown how to extract the information required to satisfy both the primary and secondary objectives.

COMPUTER SIMULATION

The key to the simulation is in generating random errors which are unique to each experimenter during each run of each experiment. An experiment consists of any number of runs selected by the experimenter. A run consists of one set of independent variables. This simulation is implemented on a Hewlett-Packard 9830 desk calculator. This calculator, as do others, has a built-in random number generator which will generate a uniform distribution of random numbers. The uniform distribution can be transformed mathematically to a normal distribution with any desired mean and standard deviation. The problem is that each time the sequence is started the same sequence of random numbers is generated unless the random number generator seed is changed.

In order to circumvent this situation in the simulation, each experimenter is assigned a user number and each session at the calculator is an experiment consisting of n-runs. The user is required to number his experiments and runs sequentially so that experiment 1 consists of runs 1-n, experiment 2 starts with run n + 1, and so on.

The seed for the random generator is computed from the product of the user number and the experiment number so that the random errors will be unique to each experimenter during each experiment. A flow chart for the simulation is shown in Figure 1.

As can be seen from the flow chart two traps are provided. The first to insure that the variables entered by the experimenter are within the ranges specified by the problem statement. The second is to insure that the errors generated do not exceed the bounds set by the instructor.

Figure 2 shows the computer display and experimenter's input for a typical interactive session. Figure 3 shows data generated in a typical experiment. The dependent variables are computed from any functions the instructor cares to insert into the program.

DISCUSSION

The simulated problem may be visualized as existing in a dimensional space. For example, the problem of Figure (3) can be considered as a set of points in a cube whose three axes are the three independent variables, as illustrated in Figure (4). This cube is therefore the test space from which a pattern of points may be chosen to constitute a test plane. A good strategy is to choose extreme points of the space; corner points, for example. The objective of the extreme points is to determine whether or not a change in the independent variable between its extremes will cause a significant change in the dependent variables. Replication of points is necessary to determine random error; a center point is replicated to estimate curvature effects. Data values for Y_1 and Y_2 may be inserted directly into their corresponding X_1, X_2, X_3) locations in the cube, as illustrated in Figure (4) for the Y_1 variable. Once this is done, averages of Y on various planes may be run to test a specific null hypothesis, such as:

$$H_0: \mu_{HighX_1} - \mu_{LowX_1} = 0$$

$$H_1: \mu_{HighX_1} - \mu_{LowX_1} \neq 0$$

Once significant effects are determined a simple linear mathematical structure for the problem can be constructed. Details of the analysis are presented below.

ANALYSIS OF THE DATA

There are three objectives of the analysis: (1) determine which X'x affect the dependent variables when they are acting alone; (2) determine if any pairs of X's produce an interaction affect; (3) construct a linear mathematical model of the form,

$$\begin{aligned} Y &= bo + b_1X_1 + b_2X_2 + b_3X_3 \\ &+ b_{12}X_1X_2 + b_{13}X_1X_3 \\ &+ b_{23}X_2X_3 + \text{Random error} \qquad (1) \end{aligned}$$

The last three terms, containing a product of X's, are the interaction terms. Other X terms are "main effect" terms; i.e. the effect of each X acting alone.

Objectives (1) and (2) can be realized through tests of hypotheses using a simple t-test on average values of Y. Objective (3) is most easily achieved if each X is coded to a -1 to +1 range by the simple linear transformation,

$$X_{coded} = m\, X_{actual} + b \qquad (2)$$

where

$$m = \frac{2}{X_{high} - X_{low}} \qquad (3)$$

and

$$b = 1 - \frac{2X_{high}}{X_{high}-X_{low}} \qquad (4)$$

For example,

$$X_{1coded} = \frac{2}{90-0} X_{1actual} +$$

$$1 - \frac{2(90)}{90-0} \qquad (5)$$

or,

$$X_{1c} = \frac{X_1}{45} -1 = \frac{X_1-45}{45} \qquad (6)$$

The results of coding are shown in Figure (5), where average values of Y_1 data are also shown for each test point. The standard deviation of Y_1 at each location is shown in parentheses. Once Figure (5) has been constructed an analysis can proceed straight forwardly.

Step (1) Determine the degrees of freedom (df) for the data. $df = \Sigma(n-1)$, where n is the number of data values obtained at each test point. Here, df = 8 + 3 = 11, since there is 1df at each corner point and 3df at the center.

Step (2) Calculate the pooled standard deviation of the data.

$$S_p = [\frac{\Sigma(n_i-1)S_i^2}{\Sigma(n_i-1)}]^{1/2}$$

For the data of Figure (5),

$$S_p = \{[(0.41)^2 + (0.84)^2 + (1.44)^2$$

$$+ (1.22)^2 + (1.07)^2 + (2.1)^2 +$$

$$(0.90)^2 + (0.39)^2 +$$

$$3(1.13)^2]/11\}^{1/2}$$

$$S_p = 1.18$$

Step (3) Calculate a Random Error term using a two-sided t statistic. For main and interaction effects the random error term is,

$$R.E. = t_{\alpha,df} S_p \sqrt{\frac{2}{kr}}$$

where, k = number of corner points on any plane, and r = the number of replicates of each corner point. Thus, for α = 0.05 level of significance,

$$t_{0.05,11} = 2.20$$

$$S_p = 1.18$$

$$k = 4$$

$$r = 2$$

Hence,

$$R.E. = 2.20(1.18) \quad \frac{2}{(4)(2)} = 1.3$$

Curvature Random Error is calculated from,

$$C.R.E. = t_{\alpha,df} S_p \sqrt{\frac{1}{c} + \frac{1}{2kr}}$$

where c = number of counterpoint replicates for Y_1,

$$C.R.E. = (2.20)(1.18)\sqrt{\frac{1}{4}+\frac{1}{2(4)(2)}} = 1.45$$

Step (4) Test null hypotheses for main effects. This can be done by subtracting the average value of Y on a low X plane from its average on the parallel high planes. Referring to Figure (5), we have for the X_1 effect,

$$H_o: \mu_{1(X_1 low)} = 0$$

A test of the data shows that

$$Y_{1(X_1 high)} - Y_{1(X_1 low)} =$$

$$[(11.4 + 41.4 + 42.7 + 11.4)$$

$$-(43.1 + 10.2 + 9.7 + 42.4)]/4$$

$$= 0.375$$

Since $|0.375| < 1.3$, H_0 is not rejected. Therefore, from the inference provided by the data, the student may conclude that the variable X_1 has no significant effect on Y_1. Results for X_2 and X_3 show

$$\bar{Y}_1(X_2\text{high}) - Y_1(X_2\text{low}) = -1.1$$

and

$$Y_1(X_3\text{high}) - Y_1(X_3\text{low}) = 0$$

Hence, similar hypotheses on "no effect" for X_2 and X_3 and also not rejected; i.e. neither has a significant effect on Y_1.

Step (5) Test Null Hypotheses on Interaction Effects. Interaction effects are determined by comparing coverages on crossplanes. Referring to Figure (6), the X_1 X_2 interaction may be obtained by first calculating an average value of those corner points where the product of X_{1_c} X_{2_c} produces a negative sign. Similarly, an average value of Y_1 can be calculated for the plane whose corners produce a positive product of $X_{1_c}X_{2_c}$. Finally, the average value for the "negative" plane is subtracted from that for the "positive" plane. From the data in Figure (6), the X_1X_2 interaction effect is tested by the result of:

$$[(43.1 + 41.4 + 42.7 + 42.4)$$

$$-(10.2 + 11.4 + 11.4 + 9.7)]/4$$

$$= 31.65$$

Since $|31.65| > 1.3$, this interaction is highly significant.

Step (6) Calculate the Curvature Effect Ramdom Error for curvature =

$$t_{\alpha,df}\, S_p \sqrt{\frac{1}{c} + \frac{1}{2k(r)}}$$

Here R.E.C. $= (2.2)(1.18)\ \frac{1}{4} + \frac{1}{8(2)}$

$$= 1.45, \text{ for } \alpha = 0.05, \text{ 11df.}$$

$$H_o:\ \mu_{1\,\text{corners}} - \mu_{1\,\text{center}} = 0$$

is tested by comparing the coverage of corner points with the center point average. The average of the corner points is simply,

$$[42.4 + 43.1 + 10.2 + 9.7 + 41.4$$

$$+ 42.7 + 11.4]/8 = 26.53$$

Since the centerpoint average = 10.8, curvature effect = 26.5 - 10.8 = 15.7. This is highly significant because $|15.7| > 1.45$. Therefore, the null hypotheses is rejected.

Step 7 Construct an Empirical Linear Equation. Having decided which terms are significant by hypothesis testing, one can now construct an empirical equation for Y_1. Coded X's ease the calculation procedures at this point. A simple linear model for Y_1 based on coded X's, would be,

$$Y_1 = b_o + b_1X_{1_c} + b_2X_{2_c}$$

$$+ b_3X_{3_c} + b_{12}X_{1_c}X_{2_c}$$

$$+ b_{13}X_{1_c}X_{3_c} + b_{23}X_{2_c}X_{3_c} \qquad (7)$$

The coefficients b_o is simply the average of the eight corner points; $b_o = 26.5$. Each of the other coefficients is simply an estimate of a derivative such as

$$\frac{\Delta Y_1}{\Delta X_{1_c}} \simeq \frac{dY_1}{dX_{1_c}}.$$

But ΔY_1 is nothing more than the difference in averages of Y_1 between the high and low X_{1_c} planes; this difference was already calculated for each null hypothesis. Further, $\Delta Xc = +1 - (-1) = 2$, in every case.

Therefore, each coefficient of X is nothing more than $\Delta Y_1/2$. For example.

$$b_1 = 0.375/2 = 0.188, \text{ and}$$

$$b_{12} = 31.65/2 = 15.83.$$

Since only the $X_1 - X_2$ interaction was significant Y_1 would be expressed as

$$Y_1 = b_0 + b_{12} X_{1_c} X_{2_c}$$

$$= 26.5 + 15.83 X_{1_c} X_{2_c} + \text{random error} \qquad (8)$$

Step (8) Decode. Decoding is accomplished by applying the transformation directly to equation (8). Recall,

$$X_{1_c} = \frac{X_1-45}{45} \text{ and } X_{2_c} \frac{X_2-45}{45}.$$

Hence

$$Y_1 = 26.5 + 15.83 \left[\frac{X_1-45}{45}\right]\left[\frac{X_2-45}{45}\right]$$

Comments

This equation should accurately describe Y_1 over its entire range provided curvature is not significant. Since curvature is significant in this case, the equation is good only near the extremes of the tests pace. To make use of the data already obtained and obtain an accurate description of Y_1, another level of data points must be taken. The model may then include X^2 terms. However, a fit of the equation must be carried out by computer methods. Reference (1) outlines the procedure.

It should also be noted that a worksheet method can be devised from Figures (4) - (5). A worksheet allows the student to extrapolate from three to many more variables. When the number of variables becomes large, screening designs can be run using procedures essentially the same as those described here. Reference (2) describes the worksheet procedure and screening design analysis. Finally, it may be noted that coded X's play two important roles:

(a) they allow retention of a magnitude of 1 when multiplied to obtain interaction effects, as was done in Figure (6).

(b) they allow for easy calculation of coefficients, requiring only a division by 2 on data already analyzed.

CONCLUSIONS

(1) Careful test planning, based on a geometric interpretation, can lead to an easily obtainable mathematical structure for a system or component in the design phase. The simulated problem described here outlines the procedure.

(2) Once the procedure is completed, the designer may determine his optimum points by a parametric study of significant variables.

(3) If curvature is significant, then the experimenter may take more data points to obtain an accurate description of the X-Y relationship.

(4) The methodology of the geometric approach may be put into a worksheet form and extrapolated to four or more variable.

(5) Once the procedure has been established by the simulated experiment, actual laboratory experiments can be planned and executed much more effectively.

REFERENCES

(1) Geremia, J. O., "Test Less, Learn More," Machine Design, Vol. 49, No. 20, Sept. 8, 1977.

(2) Geremia, J. O. "Screen Test 1 For Product Designs," Machine Design, Vol. 51, No. 9, April 26, 1979.

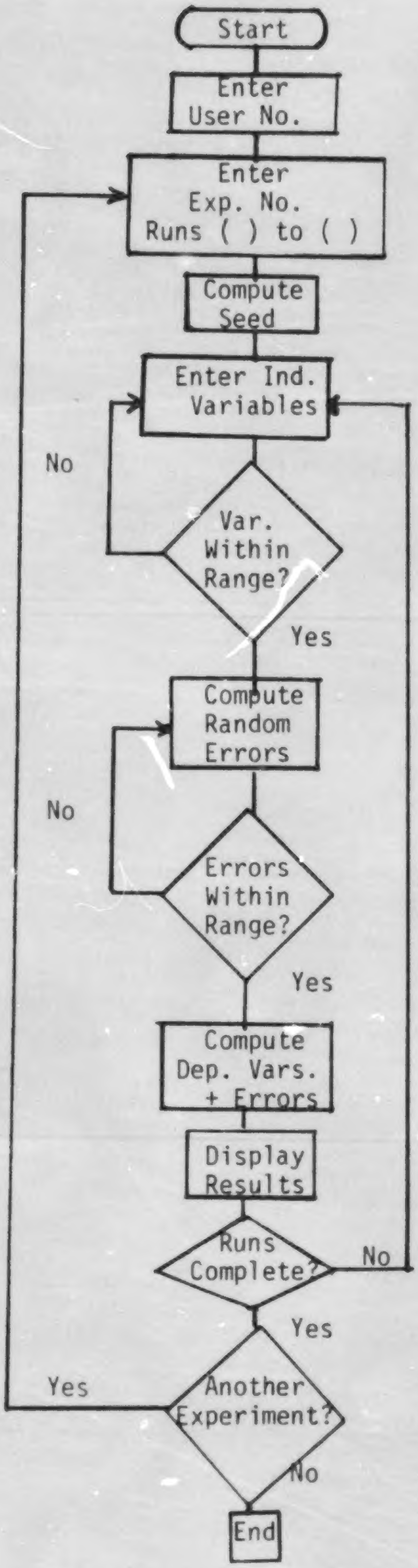

Start
Enter User No.
Enter Exp. No. Runs ( ) to ( )
Compute Seed
Enter Ind. Variables
No
Var. Within Range?
Yes
Compute Random Errors
No
Errors Within Range?
Yes
Compute Dep. Vars. + Errors
Display Results
Runs Complete?
No
Yes
Yes
Another Experiment?
No
End

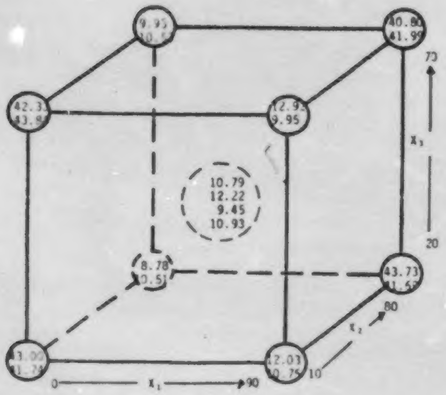
Figure 1: Simulation Flow Chart

```
USER NUMBER ?
7323967
EXPERIMENT NO.?
      :
      :
RUNS ( ) THRU ( )
1, 20
X1, X2, X3?
45, 45, 45
      :
      :
EXPERIMENT COMPLETE
ANOTHER EXPERIMENT?
NO
THANK YOU, GOODBYE
```

Figure 2: Typical Interactive Session

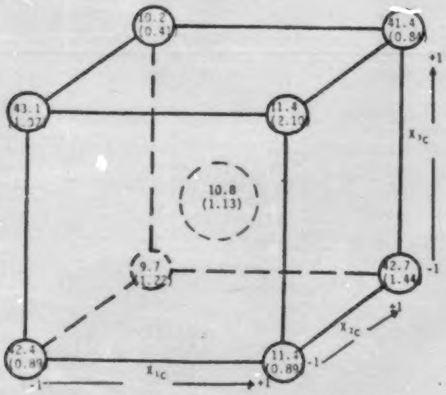
```
USER 7323967

EXPERIMENT 1

RUN  X1   X2     X3     Y2     Y2

 1   45   45     45    10.79   11.35
 2    0   10     70    42.33   18.40
 3    0   80     20     8.78   90.55
 4   90   10     20    12.03   88.98
 5   90   80     20    43.73   19.61
 6    0   10     20    43.00   19.47
 7   45   45     45    12.22    9.37
 8    0   80     70     9.95   89.93
 9    0   80     20    10.51   88.61
10    0   10     70    43.84   17.95
11   90   80     70    40.80   19.52
12    0   10     20    41.74   20.17
13   90   80     70    41.99   20.08
14   45   45     45     9.45    8.88
15    0   80     70    10.53   92.08
16   90   10     70    12.92   91.27
17   90   10     20    10.76   90.32
18   90   80     20    41.69   19.07
19   90   10     70     9.95   88.79
20   45   45     45    10.93   10.75
```

Figure 3: Typical Data


9.95
10.5
40.80
41.99
42.3
43.8
12.9
9.95
10.79
12.22
9.45
10.93
8.78
10.5
43.73
1.5
43.00
1.7
2.03
0.75
70
X_3
20
80
X_2
10
0
X_1
90

The test space consists of three mutually orthogonal axes constructed from the independent variables X_1, X_2, X_3. Here $0 \leq X_1 \leq 90$, $10 \leq X_2 \leq 80$; $20 \leq X_3 \leq 70$. Data values for $\bar{Y}_1$ are shown for their corresponding (X_1, X_2, X_3) combinations.

Figure 4.

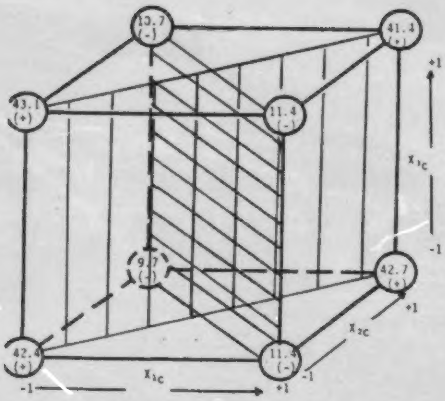

10.2
(0.4
41.4
(0.84
43.1
1.27
11.4
(2.1
10.8
(1.13)
9.7
1.22
42.7
(1.44
42.4
(0.89
11.4
(0.89
+1
X_{3C}
-1
+1
X_{2C}
-1
-1
X_{1C}
+1

For ease of analysis, actual x-values can be transformed to a coded form, X_C, such that $-1 \leq X_C \leq +1$. The simple linear transformation is obtained from

$X_C = mX_{actual} + b$, where $m = \frac{2}{X_{high} - X_{low}}$ and

$b = 1 - \frac{2\,X_{high}}{X_{high} - X_{low}}$. X_{high} and X_{low} are the extreme values of the actual X's. Also, average values of Y can be substituted at each location. Here averages for Y_1 are shown, rounded to the nearest tenth. Values of the standard deviation of Y_1 for each condition are shown in parentheses.

Figure 5.


10.7
(-)
41.4
(+)
43.1
(+)
11.4
(-)
9.7
(-)
42.7
(+)
42.4
(+)
11.4
(-)
+1
X_{3C}
-1
+1
X_{2C}
-1
-1
X_{1C}
+1

Interaction Effects are defined on cross-planes of the test space. For the X_1-X_2 interaction the positive cross-plane is constructed by connecting corners where the product of X_{1C} X_{2C} is + 1. The negative plane is defined by those corners where X_{1C} X_{2C} is - 1. Average values of Y_1 are shown inserted at each test point.

Figure 6.

444

BUILDING AN EFFICIENT AND EFFECTIVE MANAGEMENT INFORMATION SYSTEM

Jon C. Marshall
Associate Professor of Education
School of Education
University of Missouri-St. Louis
St. Louis, Missouri 63121

ABSTRACT

Presented in this paper are the characteristics of two M.I.S. using optical scan forms for data entry. The longest standing system is designed to register and track participants of the Educational Opportunity Center (EOC) for Metropolitan St. Louis. Client data are entered using "Information" and "Action" forms. Participant records are stored in direct access files. Mark IV is used to generate the files and subset information for month, quarter and annual reporting. Data turnaround is usually 10 to 15 days.

The second M.I.S. is designed to manage data for a large Teacher Center Staff Development Program. Initial entry into this system is through any of three needs assessment forms that are optical scanned. Participation in inservice programs is tracked using an Evaluation Form that is program specific. The keyed information for these forms is generated from permanent Presenter, Program, Objective and Topic files using Topic and Objective Header forms. Completed Evaluation Forms are periodically scanned and entered into the participant data files using Mark IV. Monthly accounting reports and semi-annual and annual summary reports are generated.

Both these systems illustrate the use of optical scanning in developing and maintaining participant information files that can be utilized to track participation and assess program effectiveness.

INTRODUCTION

One of the perennial problems of educational administrators and program coordinators has been the management of information for individual, program and system assessment and planning. The most frequent mode of information storage has been in individual accumulative "paper" files maintained at the staff level, with summary files at the program and system levels.

The efficiency of this process has always been a problem due to inordinate amounts of paper work needed to maintain the files; the excessive amounts of staff time often needed to periodically summarize the data; and the usual time lag between data collection and administrative utilization. At best, this has been a costly procedure, with limited amounts of factual information reaching the program and system decision-makers on a timely basis.

The advent of computer processing provided hope that this cumbersome data handling problem could be alleviated. In many cases this has happened. In many others, however, the problems are still just as bad, if not worse. A couple of examples will serve to illustrate the problem.

Example 1 - At a local university, the various colleges and divisions have over the years developed their own systems for processing by hand enrollment information during student registration. In most instances, administrators had current enrollment figures available to them within a few minutes of actual registration. Thus, they were able to monitor enrollment patterns and, at a moments notice, adjust offerings to meet students' needs. In 1979, this system was computerized, with on-line data entry. Due to the centralization of data handling, failure of data processing personnel to comprehend administrators' needs, lack of sufficient programming to generate the reports of interest, and other critical factors, the enrollment data are now available to the administrators within one week of registration. Many of the administrators have re-initiated their own systems to meet the critical immediate data needs, thus requiring double entry of the information. Due to inadequate planning, funding, and understanding of data needs, the computerized M.I.S. has proven to be "non-response" to the decision-makers needs.

Example 2 - An area state department agency handles a statewide multi-million dollar service delivery program for individual clients. At any given time, field services may be delivered to between 4,000 and 5,000 clients. Field agency reimbursement is by unit of service delivery. Thus the state department must maintain a service delivery data base to include individual identification, field agency identification, and monthly units of service delivery. The individual identification is necessary to determine system abuse through claiming duplicate services. However, to order to assess program trends and adequacy of service delivery, several other bits of information must be entered into the system. The M.I.S. has been designed for central-agency keypunch data entry. After one year in operation, the agency was over six months behind in keypunching, making it impossible to run any of the needed analysis reports and forcing emergency allocations for field agency reimbursement. A sister agency earlier had designed a similar system to track individual participants using field agencies it certified. After several years in operation, it still required over six months after the end of year to obtain the initial run of the participant report. Both of these agencies were considering direct on-line data entry from field sites. This process had been slow in developing due to initial equipment and programming costs, maintenance costs, and problems in data verification.

AN ALTERNATE

A method for data entry that can alleviate the time pressures and data verifications problems noted in the second example is optical scanning. In making this suggestion, it is realized by the author that many individuals have experienced extreme difficulties with scanning, and that many others view it as "second-class" to direct on-line data entry.

For those persons recounting horrible experiences with scanning, two things can be stated. First, in recent years, the scanning equipment has improved greatly. Utilizing an NCS scanner, I have witnessed very little difficulty in obtaining rapid, accurate data turnaround, even with prefolded mail surveys and multiple-page test booklets. Second, in interviewing individuals who reported negative experiences with scanning, two primary factors became obvious. For the most part, they assumed that the utilization of scanning forms would take care of data entry and verification concerns. Obviously, they became disenchanted with the process. For accurate data entry, one must husband the data forms prior to scanning and data verification procedures must be built into the scanning programs. As might be anticipated, the other factor relating to negative experiences pertained to the personnel. A scanning operation is no better than the quality of the staff running it!

For those persons viewing scanning as "second-class," it will suffice to say that for large scale, repetitive data entry, more data can be entered much more quickly through scanning than keypunching and usually much more economically.

Prior to deciding on the data entry method(s), an administrator should examine his/her needs and options carefully. Such questions as the following should be asked:

- What data do I need/want to maintain?
- What data do I need to be reported? In what form? And, at what time intervals?
- Do I need to interact directly with the data file? Or could some or all of my reports be batch processed?
- How could I obtain the data?
- What methods are available for data entry? How would these methods effect my collection methods? And, how would they effect reporting?
- What are the cost factors associated with each option?

If "large" amounts of data are to be maintained on each individual; if direct, interactive data entry is not necessary; if there is to be a continuing repetitive sequence of data entry; and/or if, in large scale data entry, costs need to be minimized, optical scanning could be a viable alternative.

TWO OPTICAL SCAN M.I.S.

Educational Opportunity Center (EOC)

EOC is an educational-social program designed to recruit low-income individuals into post-secondary educational programs. The program in St. Louis is in its eighth year. For the first four years, a team of individuals were retained to monitor the program through obtaining client information from individual field centers, analyzing the information and reporting to the program director and steering committee. The information was hand collected, key punched, computer analyzed, and presented in written reports.

During the fourth year, an optical scan data entry M.I.S. was built. The field staff completed and Information Form on each new participant and an Action Form each time a participant was seen (see Chart 1). Placement followup also was recorded on Action Forms.

This M.I.S. is currently a production system. Once a month their forms are mailed in for processing. Each form is proofed and needed corrections made. Then, the forms are scanned. During scanning, the ID, date range and Center number are checked for reasonableness. Any reject forms are corrected and rescanned.

The data from the Information Forms are loaded into the file creating new data records. The data from the Action Forms are matched (by ID) against the data records in the accumulative file. All matched records are added to the file and the ID's from the non-matched input records listed. The non-matched Action Forms are examined by hand to determine whether the individual is not in the data file or the ID is miscoded. All corrections are made, the sheets re-scanned, and data entered on the file. The data file is developed using Mark IV.

After all data are on the permanent file, the information needed for reporting is generated using a temporary Mark IV subset file. PL1 programs are run against this file to generate the management reports. Reports are generated for each month, quarter, and year. All subset files and subsequent reports are generated for specified data ranges.

The reports generated for EOC include, for each center and overall program, summaries of participants by age, etho-racial classification, income and other background characteristics; by services provided by counselors; and client placement and follow-up. Reports are usually delivered to the program director by the 15th of the following month for the previous month, quarter or year.

Staff Development Program

The Staff Development Center provides inservice programs for teachers and administrators in a large metropolitan school district. The program is in its seventh year of operation. For the first four years, most of the data handling was done by hand by the center clerical and professional staff.

During the third year of the Center, the needs assessment instrument was developed into an NCS scannable booklet. Subsequently, two other needs assessment instruments were similarly developed. These instruments served as the entry base for building the M.I.S. since they included individual identification and needs information. Data from these instruments are reported to the Center staff for use in program planning.

The M.I.S. is currently a production system. It has two primary components. The first component is in the generation of Evaluation Forms that are completed by inservice program participants. The forms serve the dual purpose of obtaining evaluation information and tracking individual participation. To obtain the forms, the Center secretary completes an Objective Form and Topic Form for each planned program. Once a month these forms are batch processed printing specified numbers of Evaluation Forms for each program. Printed on the forms are the program objectives, topic area, new topics of interest, program title and program number. The objective and topics are pulled from an objective bank and Topic Bank, respectively. The Objective Bank can include over 5,000 entries and the Topic Bank over 500 entries.

The participants attending an inservice program complete the Evaluation Forms. Included on the form for rating are the objectives, topics, and program characteristics. In addition, there is space for up to 20 item pre- and posttests and the participants' ID. After completion of the program, the Center secretary codes each form for the number of hours attended and the hours for reimbursement.

Once a month, the forms are processed, using the Objective Form and Topic Form as header sheets. An SAS accounting program is run against the data and a Reimbursement Report prepared. The data are then entered into the accumulative file, by individual participant. Using any specified data range, a data file is generated using a Mark IV subsetting routine. An SAS routine is used to generate summary reports by program, presenter, program objective, program topic, and participants' home schools. Unduplicated and total counts of participants, rating summaries, and t-tests for pretest-posttest changes are provided in the reports.

CHART 1: M. I. S. FOR THE EDUCATIONAL OPPORTUNITY CENTER

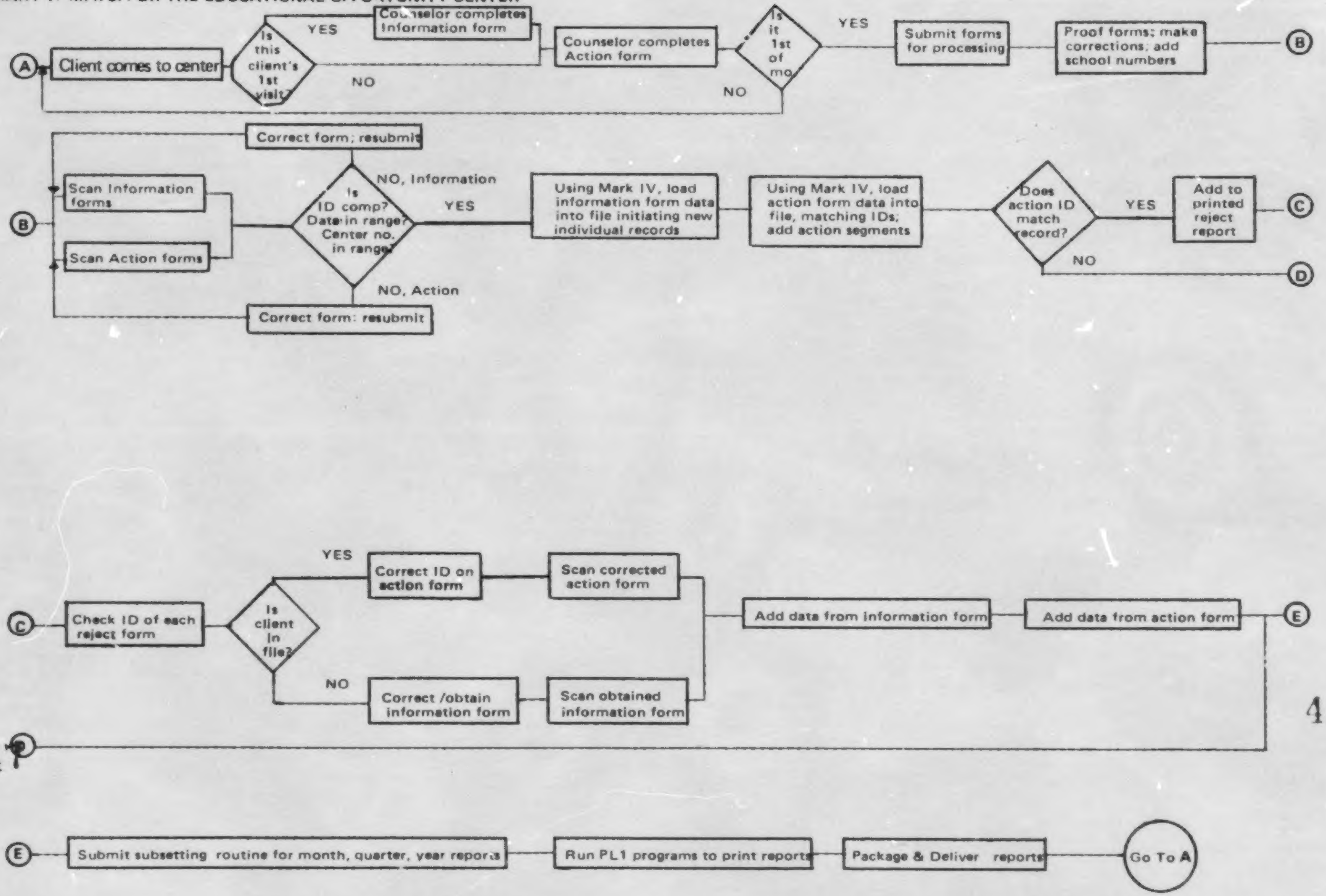
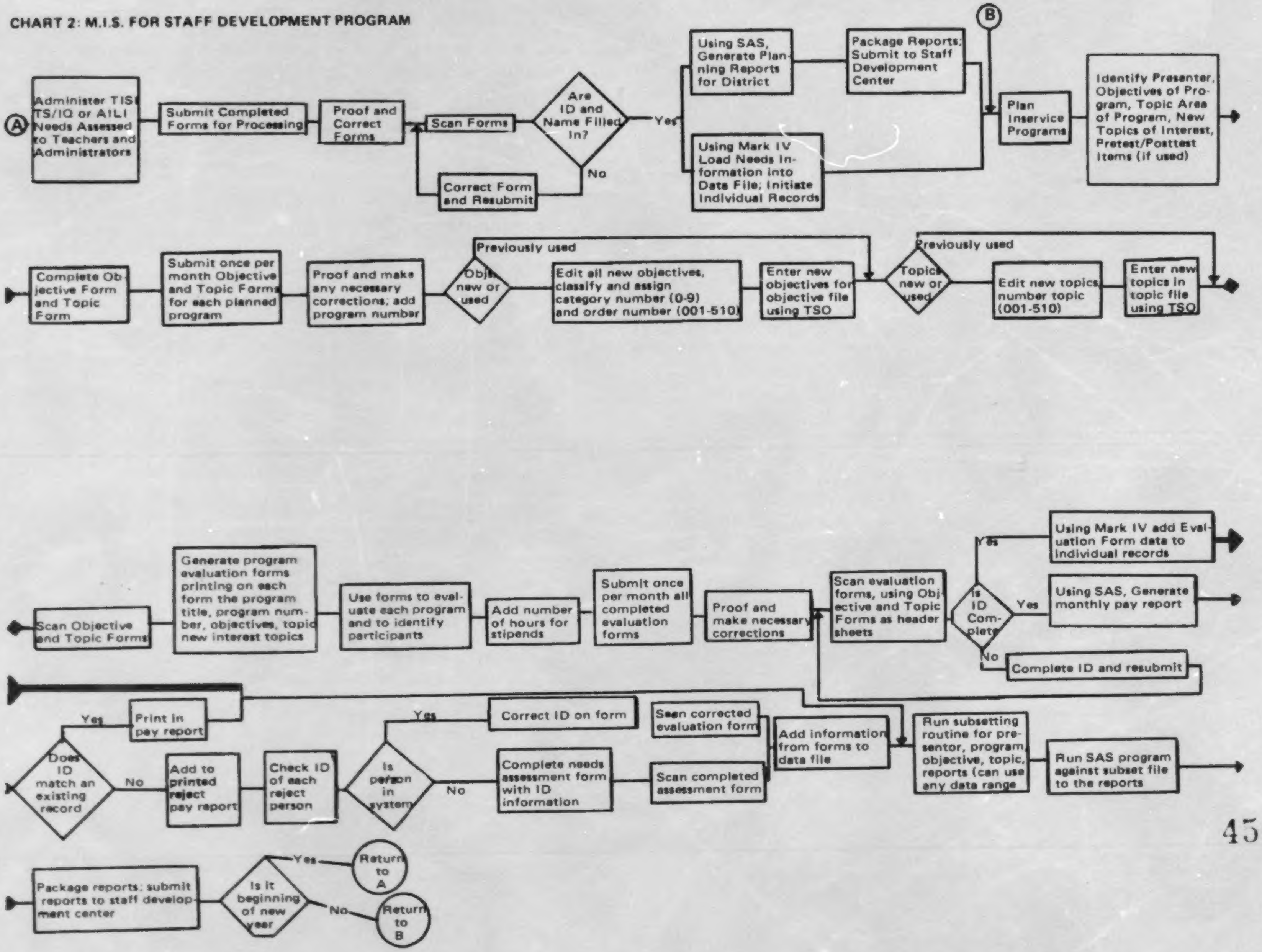

A
Client comes to center
Is this client's 1st visit?
YES
NO
Counselor completes Information form
Counselor completes Action form
Is it 1st of mo
YES
NO
Submit forms for processing
Proof forms; make corrections; add school numbers
B
B
Correct form; resubmit
Scan Information forms
Scan Action forms
Is ID comp? Date in range? Center no. in range?
NO, Information
YES
NO, Action
Correct form: resubmit
Using Mark IV, load information form data into file initiating new individual records
Using Mark IV, load action form data into file, matching IDs; add action segments
Does action ID match record?
YES
NO
Add to printed reject report
C
D
C
Check ID of each reject form
Is client in file?
YES
NO
Correct ID on action form
Scan corrected action form
Correct /obtain information form
Scan obtained information form
Add data from information form
Add data from action form
E
D
E
Submit subsetting routine for month, quarter, year reports
Run PL1 programs to print reports
Package & Deliver reports
Go To A

CHART 2: M.I.S. FOR STAFF DEVELOPMENT PROGRAM


A
Administer TISI TS/IQ or AILI Needs Assessed to Teachers and Administrators
Submit Completed Forms for Processing
Proof and Correct Forms
Scan Forms
Are ID and Name Filled In?
Yes
No
Correct Form and Resubmit
Using SAS, Generate Planning Reports for District
Package Reports; Submit to Staff Development Center
Using Mark IV Load Needs Information into Data File; Initiate Individual Records
B
Plan Inservice Programs
Identify Presenter, Objectives of Program, Topic Area of Program, New Topics of Interest, Pretest/Posttest Items (if used)
Complete Objective Form and Topic Form
Submit once per month Objective and Topic Forms for each planned program
Proof and make any necessary corrections; add program number
Obj. new or used
Previously used
Edit all new objectives, classify and assign category number (0-9) and order number (001-510)
Enter new objectives for objective file using TSO
Topic new or used
Previously used
Edit new topics number topic (001-510)
Enter new topics in topic file using TSO
Scan Objective and Topic Forms
Generate program evaluation forms printing on each form the program title, program number, objectives, topic new interest topics
Use forms to evaluate each program and to identify participants
Add number of hours for stipends
Submit once per month all completed evaluation forms
Proof and make necessary corrections
Scan evaluation forms, using Objective and Topic Forms as header sheets
Is ID Complete
Yes
Using Mark IV add Evaluation Form data to individual records
Yes
Using SAS, Generate monthly pay report
No
Complete ID and resubmit
Yes
Print in pay report
Does ID match an existing record
No
Add to printed reject pay report
Check ID of each reject person
Is person in system
Yes
Correct ID on form
No
Complete needs assessment form with ID information
Scan corrected evaluation form
Scan completed assessment form
Add information from forms to data file
Run subsetting routine for presentor, program, objective, topic, reports (can use any data range
Run SAS program against subset file to the reports
Package reports; submit reports to staff development center
Is it beginning of new year
Yes
Return to A
No
Return to B

OUR ADVENTURES WITH THE COMPUTER

Jane McClellan, Curriculum Coordinator
Holley Griffin, Teacher of Computer

In a decade when all are crying "back to the basics", it is not easy to look ahead, apply foresight, brace the tide, and forge into a computer program.

Fort King's principal, Ken Vianello, had the necessary foresight; our curriculum coordinator, Jane McClellan, had the tenacity; and best of all, our teacher, Holley Griffin, was ready to take on the mechanical wizard. The only ingredients missing for a successful start were money, classroom location, curriculum and scheduling. All surmountable obstacles, we felt.

We approached the computer industry for financial help, and to their credit they were interested in the idea but---. We tried grant writing and made the commendable stage but not the money level. Only minor setbacks!

Since outside sources of funding were not forthcoming, we looked internally. Our school is located in a middle class neighborhood in a city of about 45,000 people, a farming, horse raising county in mid-Florida. Our parents are interested in their children and the quality of the education they receive, thus encouraging them in academic endeavors. As a result of their interest the community enthusiastically supported a sale of spices with a computer lab as the goal. The total school-community effort added an excess of $10,000. The first impediment was gone.

While the sale was in progress, a school committee investigated all possible microcomputers, attended every possible display, and read brochure after borchure. We hunted for a secure, accessible location in the school for the classroom. We devised schedules to make the most use of the machines. We gathered books and magazines to start a curriculum.

Our school plant had two teaching auditoriums less frequently used when we changed from junior high to middle school. Our school system's maintenance department redid one into a pleasant, workable lab. The floor was built up and carpeting added to help with the sound. The old tables were adapted to hold the micros. Impediment two was gone - we had a classroom.

In order to give all eighth grade students satisfactory access to the lab, a schedule was developed which would place them nine weeks in the computer lab and nine weeks in a reading lab each semester. This gave the teacher five classes of approximately thirty-two students each semester.

The most complicated time consuming procedure was still incomplete. From the very beginning, thought, time, research, development and rejection went into creating a curriculum design. Up to this time, all manuals and material had been written for individuals who were intrinsically motivated computer lovers. We were facing a typical heterogeneous group of eighth graders with reading levels from third to eleventh grade, with all the attributes of an adolescent and with a need to view computers as relevant to every day school life. Working as a unit, the school advisory committee, administration and faculty established goals. We wanted complete academic coverage, not just math. We wanted the students, according to their ability to interpret language between the computer and themselves, to labor with the limitations of both, to organize material into logical sequential steps for program implementation and to create programs. We wanted students to view the computer as a challenge, as a study aid and as a potential for the future. Did we dream?

Our vision of a marriage between computer literacy and programming skills emerged. The literacy, we perceived as necessary to all - the ability to read and comprehend material on and about computer is a skill of an intellectual being of the '80's. The literacy alone was like basic math facts without any application. We saw programming skills as the completion of the package. Surely, our responsibility was to offer both to our students. We had committed ourselves to these goals when the first group of eighth graders entered the door.

By May 1979, we had provided fifteen TRS 80 Level II, 4K machines, a converted teaching-auditorium, a schedule of eighteen weeks on a nine week rotation for all three hundred eighth graders and a curriculum plan. The whole scene was tentative but interesting.

Ms. Griffin had arranged lessons to introduce the machine and a little history but she found kids wanted to touch. By the second day she illustrated a few commands and off they (the students) went. About half of each class stayed with the teacher for instruction and the rest worked on. They used the TRS-80 manual, taped instruction and trial and error. In nine weeks time their CRT was printing OM ERROR on the screen. More frustration for administration! By the second nine weeks we had to upgrade all machines to 16K.

Fort King is a very effective middle school. Our schedule is student oriented. We work on a seven period day, providing a schedule of mathematics, English, social studies, science, computer/reading, physical education, and pre-vocational exploratory. Since our plan left the lab open two periods, the teacher of the gifted student program was scheduled to work with his seventh grade students a period a day. The final "open" period was used by Ms. Griffin for small group guidance activities to do problem-solving puzzles. It is also a period which leaves a few machines open for teachers to come in to learn or to work.

For two semesters our lab was used by the community college for classes. The college professor and his adult students were most helpful to us and did much to further our image in the community. Two retired gentlemen from the class, computer buffs, were so enthusiastic about our efforts, they submitted an article on our progress to "80 Microcomputing" and we were in print. The magazine article gave us many requests from around the states for our curriculum. Ms. Griffin took that first summer to write down her ideas, to organize her activities and to build a curriculum framework. A publication, "A Byte of Computer Literacy", was the result of her efforts.

Our biggest pluses were our faculty and our students. The teachers themselves evidenced much interest in the program. They came in to the lab on planning periods and before school. They encouraged Ms. Griffin in her efforts and supported her in the community.

As schools do, we increased in size from three hundred eighth graders to three hundred ninety-three eighth graders. The school year 1980-81 presented us with new decisions to make. Were we going to exclude some eighth graders from the computer program? Could we schedule more classes into the already existing lab? Should we shorten the time in lab to allow for more classes? Should we introduce another rotation? We had more questions and we needed answers.

Our evaluations had all indicated the students felt the time was too brief. Ms. Griffin was concerned about less time in the curriculum. A schedule of three rotations of just twelve weeks each left us with reading, computer, and ? . What were the alternatives? We had another faculty member with some course knowledge of BASIC!! She was willing to take on a mini-lab of five TRS-80 level II 16K's. We only make a mistake once.

Mrs. Hancock was also certified in

reading so she took on a schedule of five reading classes for nine weeks and the same five classes continued into computer lab. To further complicate her program she had three eighth grade classes and two seventh. More decisions?

Were we ready to start seventh graders in the program? Again, together, we decided "no." The gifted eighth graders presented some problem in regular computer lab classes because they were accustomed to a one on one with the machine and thus proceeding at a rapid individual pace. This problem needed solving before we took on another. We determined to run Mrs. Hancock's seventh grade classes as a math lab and use CAI. Several students, of course, wanted to know "how." So manuals were made available for those who wanted to go on. The school soon had two classrooms operating from 7:25 a.m., when the first bus arrived, until 3:00 p.m., when the last bus left.

The school year 1980-81 brought other changes. In January Fort King welcomed a new administrator, Mr. Frank Washington, Jr. Already impressed with the venture, he gave it full support as we explored expansion for 1981-82. For color and graphics, and for a further challenge to our advanced and/or motivated students, we added six Atari 400's to our lab. This allowed more TRS-80's to be placed in the mini-lab, four Shacks to be placed in business education to provide a computer station there, one Atari to be placed in the media center and one in an academic area.

Mathematically capable sixth grade students are in a program of independent computer study. They come to the library and work for their math period The students are most enthusiastic about the project and their parents are pleased that the school has challenged their children. C.A.I. is used by Mrs. Strader, a 7/8 English teacher who puts her weekly spelling words into the computers on Monday and allows students to practice on the micro. We plan to keep records to check improvement in grades and attitude. The spelling program we use was written by Ms. Griffin. With the increased use of micros in homes, we lost our fear of handling seventh graders in the regular program so this current year there are eleven eighth grade classes and two seventh grade classes using the lab approach.

We have tried one very interesting experiment. Ms. Griffin wrote a program introducing Haiku to students and giving instructions on creating these three line Japanese poems. Mrs. Strader agreed to teach her classes using the same format and examples as Ms. Griffin. Poor Mrs. Strader spent thirteen times as long checking papers and counting syllables. She argued in favor of the oral teaching though because she felt her students' original efforts would reflect more of the true feeling of Haiku. Both groups were tested. Their rote learning showed litte difference. A larger percentage of students from the computer classes passed the post test, but the students from regular class made higher scores. Other language arts teachers read the students' original efforts and ranked them. Among the top twenty-five, thirteen came from Mrs. Strader and twelve from Ms. Griffin. Conclusion?? It may be if you're a good student, you're good. But it sure gave Ms. Griffin more time to create new programs!!

To date, maintenance problems and down time have been minimal. Our school maintenance personnel help with simple problems. Shack and Sound Ideas (Atari Dealer) have been cooperative. The students themselves have been remarkable in care of the machines. While they are not assigned a place, they adopt a machine after the second day in class and go to it. It is theirs, therefore if it "goes out", they are out of commission, too. Their enthusiasm for the activity lasts the full nine weeks so care of the machine and material is automatic.

Both teachers, Ms. Griffin and Mrs. Hancock, require original programs from each student. The assignment contains specific "musts" and is dependent on student ability. The students "play" each others' efforts even though the programs may be easy math or language drill or a complex bowling game. "Bowling" was created by Jon Seymour, a 1980-81 eighth grade student. It is an outstanding effort. Jon is very proud of his achievement and, as many authors, he is not satisfied with the program and continues to work to enhance it. He gave us permission to print an introduction here for your perusal.

```
0 / BOWLING  BY JON SEYMOUR
2 GOTO5
3 OUT255,6:OUT255,O:RETURN
5 RANDOM:CLEAR250
9 DIM S1(12,4),S2(12,4),P(12,4),FS(12,
  4),F(12),PS(12),SC(10),H$(10),CR(6)
11 GOSUB4500:CLS:PRINT@476,"BOWLING":
   FORT=;TO1000:NEXTT:T=158335
12 POKET,66:POKET+1,128
13 IFINKEY$○""THEN50:ELSEFORV=1TO20:
   NEXT:T=T-1:IFT○15812THEN12
14 POKE15838,128:POKE15777,87:FORT=
   1TO300:NEXT
15 POKE15839,128:POKE15905,76:FORT=
   1TO300:NEXT
16 POKE15840,128:POKE15717,73:FORT=
   1TO300:NEXT
17 POKE15841,128:POKE15845,78:FORT=
   1TO300:NEXT
18 POKE15842,128:POKE15973,71:FORT=
   1TO1000:NEXT
19 T=15813
20 POKET-1,128:POKET,66:IFINKEY$○""
   THEN50:ELSEFORV=1TO15:NEXT:T=T+1:
   IFT=15847THEN35
25 IFT=15837THENPOKE15837,128
26 IFT=15841THENPOKE15777,128:POKE
   15905,128
30 IFT=15845THENPOKE15717,128:POKE
   15845,128:POKE15973,128
33 GOTO20
35 FORT=1TO500:NEXTT:PRINT@486,"BY":
   PRINT@546,"JON SEYMOUR"
37 FORT=1TO1500:NEXTT
40 CLS
```

As the program continues, it permits two to bowl a full ten frames and the extra if needed, shows strikes and spares, keeps and displays scores, and even accumulates totals to show you the highest score recorded.

For the past three years Fort King has continued to forge ahead - not only providing awareness and programming skills, but also implementing programmed courseware in other areas of academic concern. Students use CAI in language arts, math, and social studies. They, themselves, assist teachers by developing courseware for specific needs. As we continue to advance in our use of micros in education, our students, parents, and faculty see new uses. We are just beginning!

A certain future step in our venture is some management system for administrative and teacher use. Our basic areas have a clearly defined set of spiraling goals. In conjunction with middle school philosophy of encouraging each individual student to achieve his maximum, the management end of instruction is time consuming. As commercial software becomes more sophisticated, we hope to adapt it to our program. This should provide us with a management system unique to our needs.

The future? "The future is now with computers," as one of our eighth graders told us. And we at Fort King Middle School strive to remain one step ahead of the NOW!!

FORT KING MIDDLE SCHOOL
MARION COUNTY
545 N. E. 17TH AVENUE
OCALA, FLORIDA 32670

COMPUTER EDUCATION -
A PRELIMINARY PLAN OF ACTION
For the Dade County Public Schools

-Alan T. Olkes

Computer literacy is fast becoming a necessary and basic skill for survival in today's technological society. Although computer education, undoubtedly is now becoming accepted by the education community as the "new" basic skill, proponents of this new basic skill have started calling the lack of computer literacy "the next great crisis in education." Despite their integral incorporation into everyday life, computers have had very little impact upon the curriculum in most school districts. Most students graduate from our schools as computer illiterates, unaware of the capabilities or limitations of computers. They do not know how to use the computer as a tool, nor are they even aware of its effect on their lives.

In Dade County the current curriculum did not adequately reflect the rapid and overwhelming influence of computers on our present and future lifestyles. Efforts were needed to incorporate computer literacy into the curriculum at all levels and in varying degrees of complexity, depending on student and teacher interests and needs.

It was the purpose of this report to present a plan of action in order to establish computer literacy as a basic skill in Dade County Public Schools. The report draws on and incorporates the recommendations of the Microcomputer Committee established by the Bureau of Education in November, 1980. It is part of a continuing investigation by school district personnel into the whole area of computer education as a basic skill from kindergarten through the adult program.

A pilot effort, initiated in 1979 through Title IV-B, put twenty microcomputers in various junior high schools. Senior high schools had begun purchasing microcomputers on their own. There were even several elementary schools independently pursuing some sort of computer-related educational experience. The Bureau of Education Microcomputer Committee determined that there were approximately 250 microcomputers in the Dade County Public Schools, the majority housed in senior high schools.

The committee concluded that districtwide guidelines, policies, and procedures for the purchase and use of microcomputers were needed in order to ensure that they are used to their maximum and with the greatest cost-effectiveness. The uncontrolled and unplanned acquisition of microcomputers was not in the best interests of economy nor did it facilitate the development of districtwide inservice plans and curricular offerings.

The Microcomputer Committee's subcommittee on equipment selection procedures adopted the Thomas Selection Model, with minor modifications, as its basic planning guide. This model is particularly appropriate to the instructional situation and provided a useful guide for committee analysis of existing options.

The subcommittee determined that the four uses of prime importance were:

°Computer literacy
°Computer Assisted Instruction (CAI)
°Computer Managed Instruction (CMI)
°Computer programming

Once the prime uses of microcomputers were determined, it became necessary to determine "computing requirements." Technical information was compiled from a variety of sources including manufacturer's specification sheets, professional journals, data processing professionals, independent evaluations and feature comparison charts. The desired features and support were then discussed and compiled into equipment specifications. The subcommittee considered educational requirements to determine physical features listed in the specifications.

The committee met with industry representatives and provided them with background information about their work, tasks of the committee, and the responsibilities and relationships of the various subcommittees. The committee sought and received input from the vendors regarding the planning process and special considerations for specification development.

Following this, the committee scheduled a meeting to examine microcomputer equipment and software, review specifications, exchange opinions, and rate the microcomputer systems based on the specifications, presentations made, commercial literature, feature comparison sheets, and available software. The major microcomputer manufacturers were invited to participate in this equipment demonstration. Ten major companies agreed to demonstrate; however, only seven demonstrated as scheduled. The committee examined and ranked machines by comparing machines against specifications, comparing various features, and examining software available.

Apple, Atari, and Bell and Howell, who had received high committee evaluations in the first demonstration, were invited back for an in-depth look at equipment and software. Bids were then requested from these three companies.

A decision was made to accept Atari's low bid and to recommend purchase of these machines to the School Board for use in the initial phase of Dade County's computer education program. At the conclusion of the initial phase, a decision will be made as to whether additional bids should be solicited.

It appeared evident that several configurations for the use of microcomputers should be established for 1981-1982. Funding for the equipment involved was available from Title IV-B and adult general funds; however, personnel costs needed to be met through regularly funded local sources.

The model envisioned represented a broad spectrum of possible configurations and uses, and is planned to serve the dual purpose of introducing technology to a wide variety of potential users while serving as a catalyst in encouraging further expansion. It is not considered the ultimate situation, but rather the first step in an extensive evaluation procedure.

The initial plan centers on four (4) feeder systems, and the provision of microcomputers to several adult/vocational education centers and programs. It will result in the establishment of microcomputer laboratories in most senior high schools, adult centers, and technical education centers in the County, with additional installations at selected elementary and junior high schools.

The feeder system pilot schools will serve as a testing ground for the development of a comprehensive computer education articulation plan for the whole of Dade county. Within these schools the major issues of: hardware, software, instructional uses, and teacher/administrator training will be addressed. Problems will be identified, solutions will be proposed and tested, and recommendations will be made.

The establishment of these feeder pattern pilots represents the belief that comprehensive planning is the shared responsibility of administrators, teachers, and parents. Each has a vital role in the definition of computer uses, the development and evaluation of software, and the design and delivery of instructional computing programs. It also indicates a recognition that new countywide programs require a school-based trial phase.

Microcomputer labs will be established

in several adult/vocational education centers and programs. These installations represent an initial effort to develop a laboratory which not only meets basic curriculum needs relative to computer education, but also serves to each prevocational and vocational skills.

To ensure that the County's administrative and instructional staff be fully familiar with this movement, the Microcomputer Committee recommended that high priority be given to the revision of the *Master Plan for Inservice Education* to include components related to basic computer literacy as well as more advanced topics such as programming and computer science applications.

Florida International University indicated a willingness to cooperatively develop an intensive training program with the Dade County Public Schools. In addition, it was recommended that Atari, in response to its offer to provide staff development services, work with the Dade County Schools' Division of Staff Development and Accountability Programs to develop inservice components for administrators and selected teacher/trainers on the features of the hardware/software package it bid.

Further, it is recommended that the Division of Staff Development and Accountability Programs work with the University of Florida, the University of Minnesota, and other such institutions and agencies having materials specifically designed for teacher training in order to identify components, materials, and instructors to cover the following areas:

°Computer Awareness
°Learning and Teaching Basic Computer Language
°Identification and Development of Computer Software
°Vocational Applications of Computer Education
°Using Microcomputers in Secondary Science
°Using Microcomputers in Secondary Mathematics
°Using Microcomputers in Elementary Education
°Using Microcomputers in Guidance Activities
°Using Microcomputers in Secondary and Post-Secondary Business Education
°Administrative and Management Aspects of Microcomputers
°CMI Applications for Microcomputers
°CAI Applications for Microcomputers
°CAL Applications for Microcomputers
°Computer Awareness for Parents and Other Adults
°Developing Instructional Programs for Microcomputers
°Computer Applications to Special Education

A planned effort to involve manufacturers and users of computers will be undertaken during the 1981-82 school year. This vital link between the educational setting and the business sector will be coordinated through the county-level Adult/Vocational Computer Education Advisory Committee. The purpose of this involvement will be the development of programs which reflect current vocational needs and are responsive to the latest technological applications relative to computers.

A program of the magnitude being initiated in Dade County requires careful attention to the administrative and evaluative aspects. It is essential that a comprehensive, coordinated curriculum in the area of computer education be developed and that it be incorporated as a vital and functioning aspect of the K-12 and vocational/adult education curriculum. This basic skill curriculum must include provisions for varying degrees of complexity, awareness, and usage, depending on student interest and need. In order to be fully functional and effective, this curriculum must also include planned inservice activities and the involvement of school-level personnel in its development. The identification, maintenance, evaluation and dissemination of information and software need to be accomplished also. It is for this reason that the establishment of the position of Supervisor of Computer Education within the Department of Basic Skills was recommended. A person with broad knowledge of school curriculum and technical knowledge of computers is required in order to coordinate the various components of this complex and comprehensive effort.

One of the tasks of the Computer Education Supervisor will be to work with the Department of Evaluation in order to structure an appropriate evaluation design. If this initial computer education program is to fulfill its function then a clear set of objectives will need to be defined and a careful assessment of progress must be made. The appropriateness of planned outcomes must be evaluated, and the efficacy of curriculum must be measured.

A major function of the initial phase of the computer education program will be the development and fieldtesting of a computer education curriculum. This development process will of necessity be a multi-year effort. There are six areas where curriculum development must take place if a fully functioning computer education program is to be established:

°Awareness
°Literacy
°Professional Level
°Computer Managed Instruction (CMI) Applications
°Computer Assisted Instruction (CAI) Applications
°Computer Software Coordination

This summarizes a detailed comprehensive plan developed over a full year. To get copies of the full text, write to:

Mrs. Marilyn Neff
Director of Basic Skills
Dade County Public Schools
1410 Northeast Second Avenue
Miami, Florida 33132

and ask for:

"Computer Education -
A Preliminary Plan of Action"

TITLE: USING A MICRO-COMPUTER TO MANAGE LARGE FILES OF INSTRUCTIONAL PICTURES

AUTHORS: Samuel T. Price AND Larry M. Dillingham

Introduction

This paper describes and explains an operational micro-computer system for indexing, storing, identifying, viewing, and retrieving information about large files of pictures which are used during the process of teaching deaf children. The system enables teachers to perform automated searches of their picture file with the objective of identifying those useful in teaching particular concepts. After identifying a picture, the teacher may view it on a TV recorder without removing the original picture from its storage place in stand-up files. During the search process the teacher is provided with a printed list of picture accession numbers which may be used to locate and secure individual pictures from their storage places. The Picture File Management System uses an Apple II Plus Computer with 48K RAM, a line printer, and one disc drive. The video presentation utilizes a BCD Electronics Video Interface[1] and an appropriate video-tape player. The video presentation is available at the discretion of the teacher. Documentation for the new system is available from the authors.

General Design Features

In use the teacher approaches the system with an information problem. She wishes to locate one or more pictures from her large file of pictures which may be used in teaching a concept to a deaf pupil. She knows that a particular picture exists but she may have difficulty in remembering where it is filed. Since most of the pictures in the file are used to teach more than a single concept and the teacher has only a single copy of the picture, remembering how pictures are filed becomes a significant problem. For example, a teacher may use a favorite picture to teach the concepts of Christmas or run or safety. The single copy of the picture could logically be found filed under any of the three terms.

The present file management system resolves this problem by allowing the teacher to index her pictures under more than a single conceptual use. In operation the system assigns a unique accession number to each picture placed in the file. Accession numbers are used much like call numbers are used in libraries. They enable the system's clientele to actually gain physical access to a picture after its existence has been verified through a file-search procedure. After each picture is assigned an accession number the teacher is instructed to record as many conceptual uses for the picture as are needed. Conceptual uses may be viewed as index terms which are used to search the picture file. Each picture may be indexed with as many as twenty conceptual uses. The current edition of the Picture File Management System limits picture file size to fifteen hundred pictures.

The data file for the Picture File Management System consists of a series of information elements about the instructional uses of pictures. The elements are recorded on floppy discs because of that medium's greatly improved speed and convenience maintenance. Each data element on the disc begins as a data entry card which the teacher prepares. The teacher prepares one data entry card for each picture she adds to the file. After a picture is added to the data file, then its data entry card is of no further use in the Picture File Management System. The data entry cards are seen in Figure 1. The information contained on the data entry cards is all that is needed for input into the system.

The data elements contained on the data entry cards are rearranged into alphabetical order according to the conceptual uses, or index terms, under which the pictures are filed. Thus, if run or Christmas or safety were used as index terms, then the terms would be found in their appropriate alphabetical position on the data disc. When the computer finds an index term which is the object of a search, it then scans the entire data element

---

[1] BCD Associates, Inc., 1216 N. Blackwelder Ave., Oklahoma City OK 73106

DATA ENTRY CARD
(side one)

ACCESSION NUMBER 1467

CONCEPT TERMS:

CLEANING
HOME
YARD

(acc. nums between 0 - 1499)

DATA ENTRY CARD
(side two)

DATA FOR VIDEO PRESENTATION
ACCESSION NUMBER same number

MEDIA TYPE picture

VIDEO TAPE NUMBER 2

ADDRESSES ON TAPE
begin 9700
end 10400

Figure 1

associated with the term. A data element consists of two items of information. The first is the concept term itself. The second is a list of accession numbers of pictures which have been linked conceptually to that term. An example follows.

DATA FILE DESIGN

CONCEPT TERMS	ACCESSION NO'S
Christmas	10, 125, 580, 602, 345, 46, 876
run	125, 12, 365, 888, 1149, 875, 921
safety	10, 486, 125, 711, 56, 75, 876, 1102

Figure 2

The lefthand column in Figure 2 contains the alphabetical list of all conceptual use terms (index terms) used in the teacher's file. The righthand column contains the accession numbers of pictures having been indexed under each term. This arrangement is referred to as an inverted file. It is noted that number 125 appears in all three lists of numbers shown in Figure 2. This indicates that picture number 125 has been designated by the teacher as being useful in teaching all three concepts. It may also be noted that a large number of pictures have been indexed under each concept term; thus the teacher will have a selection of pictures to choose from in teaching any given concept. The logic of the search system is best explained by an actual example.

__Using The System__

A teacher approaches the system to identify and recall pictures which may be used to teach the concept of __safety__ to a deaf pupil. After she has Picture File Management System on-line by using her program diskette, she inserts her picture file data disc into the computer. She asks (figuratively), "Do I have any pictures in my file which may be used to teach the concept of __safety__?". The computer responds by finding the conceptual use term __safety__ in the alphabetical list of terms. The computer then identifies the set of numbers which are linked with the term __safety__. The list is displayed on the system monitor.

The teacher may then ask, "May I view picture number 10?". The system will respond by displaying an image of picture number 10 on the color TV monitor which is linked through an electronic interface with the microcomputer. The teacher may view all of the pictures indexed under the term __safety__. When she is satisfied, she may ask for a printed list of all the accession numbers of the pictures she has viewed. The teacher then retrieves the pictures which she needs from their physical storage place. The activity is then complete.

Other Applications

Picture files of any length may be accomodated by the Picture File Management System if a serial system of data discs is utilized. Original copies of pictures are stored, according to their accession numbers, in regular storage files. Each teacher who uses the system as a picture file management tool will require a system manual, a program diskette, and data discs. Access to appropriate hardware is also needed. After a short learning period teachers will be albe to manage picture files unassisted. A four-hour workshop is sufficient to master the system's use.

In schools having more than a single class for deaf students, or in a state residential school, a single system may be expanded to be used by any number of teachers. Such expansions are easily accomodated. Each teacher using the system would have her own program diskette and her own data discs. The system diskettes would be the same for all users. The data discs would be unique to each teacher. There would be a common data file used by all teachers and kept in a central location in the school; thus, each teacher could maintain her own picture file and also be able to search the common file. Since each teacher keeps her own data file discs, security within a cluster of users should pose few problems.

Video Presentation

It has been explained that the Picture File Management System (PFMS) provides its users with the option of viewing TV images of pictures in the file on a color television monitor receiving its signals from a TV recorder. The TV recorder is controlled in its turn by the micro-computer. A brief description of how the micro-computer is interfaced with the TV recorder follows. The reader is referred to side two of the Data Entry Card seen in Figure 1. The accession number on the card allows the teacher to call out any picture in the file for viewing. If a teacher wants to view picture number 10, she asks to see the picture with that number. The computer responds by finding picture number 10 on the materials list. Then the computer reads that picture number 10 is recorded in video form on TV tape number 2. Tape 2 is loaded on the TV recorder. Picture 10 begins at location number 9700 on the tape. It ends at location 10400. The purpose of spreading the image over 700 tape positions is that the teacher may want to scan the TV tape visually to review it. This numbering system permits that freedom. When individual pictures are to be reviewed, as they would be when searching the picture file, the TV tape is programmed to stop ten positions past the beginning of the picture. This allows the teacher to view the picture for as long a time as desired.

The video presentation is included with all versions of PFMS. The program diskette which is provided permits the user to specify whether or not the video option will be implemented. The hardware needed to implement the option was specified in the introduction.

File Management

The Picture File Management System has been developed with the idea in mind that the data files which it supports will be mainly those belonging to individual teachers or to clusters of teachers who share the same school system. The files will be personal data bases not intended to be integrated into large networks of picture files. Individual teachers will build collections of pictures and index them for entry into the data base. The PFMS software will be the same for all users of the system; but the picture file data disc, and its associated index terms, will differ from teacher to teacher. One of the design assumptions in developing PFMS is that there will be enough similarities between the words and language used by teachers of the deaf (who share physical surroundings) so that they will get similar results when they search either their own or a common picture data file. Experience will test this assumption.

The file maintenance implications of this design are noteworthy. The clientele of the system will be individual teachers who share the same school system. This means that it will not be necessary to monitor the level of integration of the various information retrieval systems which the software supports. Fragmentation of files, which is ordinarily critical does not promise to be so with this system; thus, the developers of PFMS may feel free to expand, augment and otherwise improve the software packages without fear that earlier clients will suffer from software systems which are outdated. Of even greater importance is the fact that improved versions of PFMS may be implemented in local school settings whenever they become available without fear that existing picture files will be adversely affected. The authors of the system plan to continue its development. As improvements are added they will be offered to the profession.

COMPUTER SIMULATIONS STIMULATE INDUCTIVE THINKING

AEDS 1982 CONVENTION
Orlando, Florida

EILEEN K. GRESS
DIRECTOR OF EDUCATIONAL COMMUNICATIONS

Fairport Central School District
District IMC
140 Hulburt Road
Fairport, New York 14450
(716) 223-5947

464

Simulation is a dynamic, highly complex form of instructional program. By providing an environment for students to learn decision making skills, simulations foster active learning experiences seldom available in the schools. Yet though they are widely used in industry, simulations remain the least understood form of classroom oriented computer activities. Their complexity is evidenced by the lack of clearly defined objectives for many programs. Some light can be shed on the subject by examining the major issues involved in using simulations and by suggesting an appropriate senior high mathematics application.

A computer simulation can be defined as a program that manipulates a model of the real world through the use of equations that map input values to output values. By choosing to represent only a portion of the real world, we can create a concrete situation from an abstract one. Hypotheses can be tested in an environment that approaches reality, but does not model its undersirable consequences.

The goal of modeling the real world is not to have students recall facts or develop basic skills. We err greatly when we use simulations as the primary means of accomplishing such objectives. The purpose of using simulations is to have students develop problem solving skills or to aid in their understanding of concepts. Simulations enable students to learn specific problem solving strategies by receiving immediate feedback on decisions made while in a structured problem solving environment. They can, furthermore, enable students to gain insight into concepts not normally accessible to them without the use of a computer.

Benjamin Bloom (1956) classified educational objectives in a book that is the sin qua non for educators. He defined six categories of thinking skills in his taxonomy of educational objectives. The first two can be considered higher order.

"Knowledge" objectives require the recall of facts, termindogy or other elements of a curriculum area. "Comprehension" objectives imply understanding or internalizing a concept. These lower order objectives are generally product oriented, in that the processes used to accomplish the objective are less important than the ability to demonstrate achievement of it.

High order objectives, on the other hand, are more process oriented. "Application objectives" imply the ability to apply knowledge, principles, or generalizations to new problem situations. "Analysis" requires the ability to perceive relationships between the parts of a whole. "Synthesis" goes one step further, requiring parts be put together to form a structure that was not clearly there before. Last, "evaluation" implies the ability to judge ideas, solutions, or methods.

The traditional focus of instruction is on lower order thinking skills. In a review of research studies, Dunkin and Biddle (1974) report that "...those classrooms studied have indeed shown a higher frequency of questions thought to require lower-level cognitive processes and exchanges involving knowledge in particular". The rapid rate at which new knowledge and technology are developing suggest, however, that we emphasize knowledge not for its own sake, but as it relates to higher level thinking.

Dunkin and Biddle report that teacher use of thinking levels varies directly with student use. Activities incorporating simulations foster greater emphasis on higher levels of thinking. A good program allows students to gain insight into a problem or generalize from patterns they perceive as the simulation is enacted. Inductive thinking is stimulated as students reason from specific to general cases. It is, however, the teacher's responsibility to ask the questions or state the comments that enable students to develop these inductive thinking skills. Without teacher direction, students may perceive the simulation as a game.

CREATING A SIMULATION

High level thinking skills can be fostered by either participating in a simulation or creating one. The former may be appropriate as a classroom activity in a given subject area, while the latter would be reserved for students in computer science courses. From the teacher's point of view, however, understanding how a simulation is developed can improve his/her ability to structure classroom activities that incorporate simulations.

Though simulations can be written in a game or non-game format, I shall address the latter at this time. The design process conprises seven steps, as defined below:

1. Define the problem
2. State the objectives
3. Research the problem
4. Specify the model
5. Evaluate the model
6. Write the program
7. Validate the program
8. Develop the documentation

The process can be clarified by way of an example. The number π (PI) appears early in students' mathematical coursework as the ratio of the circumference of a circle to the diameter. In junior high, students use π to compute the area and circumference of circles. At the high school level, it is again revisited when irrational numbers are explored. That π was not conceived out of thin air comes as a surprise to many students, despite its frequent application in the mathematics curriculum.

DEFINE THE PROBLEM

Classroom activities to estimate the value of π are often futile at best. Taking measurements of several circles has the advantage of being a concrete activity, but the estimates obtained are grossly inadequate. Other possible activities, including measuring the perimeters of inscribed and circumscribed regular polygons, are so complex that they mask their true purpose. A computer simulation can be developed to enable students to estimate the value of π by using a concrete example that is neither cumbersome nor overly abstract.

By employing high resolution graphics, a circle will be inscribed in a square. "Darts" will be randomly thrown at the square. The ratio of the number of darts landing within the circle to the number of darts within the square should approach the ratio π :4.

STATE THE OBJECTIVES

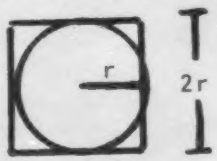
The simulation should enable students to:

1. Estimate the value of π
2. Apply basic probability laws to a problem solving situation
3. Analyze the relationship between the number of trials and the approximation of π
4. Demonstrate their understanding of ratio and proportion by explaining the model underlying the simulation

Clearly defined objectives foster effective use of simulations. The objectives should be used to determine whether the simulation is best enacted in a group or individual setting and what types of activities should be developed to accompany the software.

SPECIFY THE MODEL

We know that the area of a circle can be found by applying the formula πr^2 and that the area of a square is computed by squaring the length of a side. If a circle with a radius of one unit is inscribed in a square, the ratio of their areas is πr^2 : $4r^2$ or π:4 (see figure 1).



$$\frac{\text{Area of circle}}{\text{Area of square}} = \frac{r^2}{(2r^2)}$$

FIGURE 1

COMPARISON OF AREAS

Now if we randomly toss a large number of darts at the square (we will assume no misses), the ratio of the number of darts landing within the circle to the number of darts landing within the square should approximate the value of $\pi/4$. If we multiply by 4, an approximate value of π should appear.

The coding can be outlined by a simple iterative input-process-output structure. The number of darts to be thrown is first determined. Each "dart" is simply a randomly generated point within the boundaries of the square. Once this input is generated, counters are incremented to track the number of darts landing within each of the two figures. The ratio is computed and printed, and the process is repeated. As the number of darts increases, better approximations of π should be achieved.

EVALUATE THE MODEL

A formative evaluation is appropriate at this stage of the development process. The model should be reviewed to determine its potential adequacy for accomplishing the stated objectives. It is also important to query whether the model is an appropriate use of the computer.

Probability models for the estimation of π are found frequently in the literature. Hirsch (1977) suggests two student activities that can be used. The first involves random tossing of a needle, while the second involves random tossing of a circular "saucer". Though these models will produce close approximations to π, they are cumbersome and time consuming. Computer approximation eliminates these problems and enables students to experiment with different trial lengths.

A problem with many courseware packages currently marketed is that they are not cost effective applications of the computer. Many drill and practice programs do little more than what a file full of drill sheets can do; many simulations are not appropriate for any curriculum. A probability model of π, however, is best implemented on the computer. Neither a hands-on activity nor a pencil and paper activity can enable students to experimentally approximate the value of π as well as the computer can.

WRITE THE PROGRAM

The coding of the simulation can be accomplished by using two routines. The first uses the high resolution screen to inscribe a circle within a square. The second is the input-process-output structure for each trial. Though a professional simulation would be embellished with student directions, sound effects, and other motivational devices, this task is left for the reader. All coding has been developed for the Apple microcomputer.

Figure 2 shows a general routine for generating the circle inscribed within a square.

```
10    HOME
20    HGR: HCOLOR = 3
30    REM DRAW SQUARE
40    HPLOT 65,5 TO 215,5 TO 215,155
      TO 65,155 TO 65,5
50    R=75: REM RADIUS
60    REM CALCULATE CIRCLE COORDINATES
      USING X^2 + Y^2 = R^2
70    FOR X = 0 TO R
80    Y = SQR (R^2 - X^2)
90    REM TRANSLATE COORDINATES TO A
      CENTER OF 140,80
100   HPLOT X + 140,80 + Y
110   HPLOT X + 140,80 - Y
120   HPLOT 140 - X,80 + Y
130   HPLOT 140 - X,80 - Y
140   NEXT X
```

GRAPHICS ROUTINE
FIGURE 2

The algorithm used is based on the formula $X^2 + Y^2 = R^2$, where X and Y determine the points on a graph, and R is the radius of the circle. This equation generates a circle whose center is at 0,0. Since the center is to be placed at 140,80, lines 100 through 130 make the appropriate adjustment. For each X value generated, a point is plotted in each quadrant of the circle.

Figure 3 shows the routine for generating the trials.

```
200   REM Z COUNTS NUMBER OF TRIALS
210   REM X AND Y ARE COORDINATES
220   REM C COUNTS DARTS IN CIRCLE
230   REM S COUNTS DARTS IN SQUARE
240   VTAB 21
250   INPUT "HOW MANY TRIALS?"; Z
260   HOME: VTAB 21
270   PRINT "CIRCLE", "SQUARE",
      "(C:S) * 4"
280   FOR I = 1 TO Z
290   X = INT (151 * RND (1) + 65)
300   Y = INT (151 * RND (1) + 5)
310   DIST = SQR ((X - 140)^ 2 + (Y -
      80)^ 2)
320   IF DIST <  R THEN C = C + 1
330   S = S + 1
340   HCOLOR = 7
350   HPLOT X, Y
360   VTAB 22
370   PRINT C, S, INT (1000 *(C/S * 4)
      + .5)
380   FOR K = 1 TO 500: NEXT
390   NEXT I
```

GENERATING THE TRIALS
FIGURE 3

Though the counter S is unnecessary, it is used here for emphasis. Z allows the user to vary the number of runs. Lines 290 and 300 produce coordinates within, and including, the boundaries of the square. Lines 310-350 determine whether the point lies within or on the circle, increment the appropriate counters, and plot the point. Lines 360 and 370 position the values of the counters and the approximation of π on the twenty-second row of the screen. Line 380 is a delay loop. The process is then repeated.

VALIDATE THE PROGRAM

The completed simulation should be classroom tested to determine whether it can be used to approximate π, whether the directions are clear, and whether it accomplishes the stated objectives.

DEVELOP THE DOCUMENTATION

Documentation should include a statement of the objectives, an explanation of the model, and suggestions for classroom activities. Teacher questions should emphasize higher order thinking skills. They might include:

- What would you expect to happen if the number of trials is increased?
- In general, how many trials are needed to generate an approximation that is correct to the hundredths place?

CONCLUSION

Higher level thinking skills must be taught explicitly. The writer has attempted to give an example of a simulation that encourages such thinking. The computer is too valuable a tool to use mainly for drill and practice. Applications should be sought that enable students to comprehend abstract concepts and develop problem solving skills.

REFERENCES

Bloom, B. S. (Ed.). *Taxonomy of Educational Objectives; The Classification of Educational Goals*. Handbook I. Cognitive domain. New York: McKay, 1956.

Dunkin, M. J. and Biddle, B. J. *The Study of Teaching*. New York: Holt, Rinehart and Winston, 1974.

Hirsch, C. R. Probability and Pi. *The Mathematics Teacher*, 1977, *70*, 519-521.

VOCATIONAL TRAINING FOR HIGH SCHOOLS IN THE AGE OF THE MICRO

Paul Lorton, Jr. & Eugene J Muscat
University of San Francisco
San Francisco, California 94117

IBM is marketing a microcomputer. The personal computer has now entered the establishment and what may once have been a curiosity is now a very serious item indeed. Many educators have long recognized that microcomputers were important for education and that education could benefit from teaching with and about them. Now, with such events as the name IBM on a personal computer, that view is coming to the entire educational community - parents, administrators and teachers.

Math departments, science departments and other academic departments are working quite hard on bringing the micro in as an integral part of the curriculum whether for teaching with or about. These efforts will help prepare the college bound youth for a world permeated by computers. But what about the non-college bound student? Those students who plan on terminating their education with the 12th grade need to be given an even greater familiarity with microcomputer use than those going on to college.

Fortunately, there are many projects serving to meet this need. We will discuss some of them briefly and will indicate the major problems remaining in the effort to give appropriate vocational training to today's students whose working lives will stretch well into the 21st century.

Some Projects in Vocational Education

In every part of the country, individual teachers are bringing microcomputers into their classrooms, teaching with them and teaching about them. Starting with the first microcomputer introduced in the first math or science class, a gradual trend has been evolving over the last five years. That trend has been to place computing resources directly in the hands of innovative teachers.

When introduced into the classroom, these microcomputers allowed the individual teacher to be free of the constraints of a timesharing terminal to some distant computer. Often, these teachers had developed an awareness of what could be done with a computer and how to do it. The microcomputer gave them better control over the computing presence in their own classroom.

Many projects in vocational education grew out of this kind of effort. At the University of San Francisco, a CETA funded project to train unemployed, out of school youth for employment in the electronic office environment made heavy use of a small timesharing system to teach data entry, word processing and computer literacy. Using, as a core, the Californian Common Curriculum Units in Business Education (CCUBE), the youth were trained for employment in an office practices setting which would involve computers

469

extensively. This material has been largely adapted for use on a microcomputer. Such use of the microcomputer rather than a timesharing terminal will make the instructional experience much closer to the actual practice the student would certainly see five years from now and often would encounter today.

Data processing education is an ideal area for the introduction of microcomputers as part of the instructional environment. Those schools who have provided actual training in computer operations and programming rather than key punch training are taking advantage of the microcomputers to make the instructional experience of their students more real than is possible with a time-shared system. The student can actually work with the entire computer. As an adjunct to an introductory data processing text, we are developing a series of exercises which will use a microcomputer as the training environment for the student rather than the usual pencil and paper exercises.

The retail trade will be able to take considerable advantage of the new computing power offered by microcomputers. Computer controlled cash registers, receiving and inventory control by computer, even stock-taking with portable terminals are technologically feasible and are being used. Distributive education, while not as natural an area for microcomputer use as data processing education, is still an area which needs microcomputer literacy. As part of an ongoing effort to assist with systematic developments in specific curricular areas, we have developed several programs which are being used in distributive education. The most important part of these programs is that they help a student training for the distributive trade become comfortable with the computerized environment in which they will be working.

The availability of materials which can be used does not guarantee they will be used. Training teachers to use, construct and evaluate computer related materials for their classrooms is as important as developing the materials themselves. University courses are often a site for such training and we have taught these often. These, however, only reach teachers as students. To reach teachers in their own schools, we have begun a computer literacy course for teachers at a San Francisco high school. The specific purpose of the course is to help teachers make effective use of several microcomputers purchased for achieving ESEA Title I goals. However, the more general goal is to assist an entire high school faculty in becoming literate in the instructional uses of computers. With the participation of their peers with whom they are in daily contact, we hope to produce an environment which will foster and support sustained use of microcomputers in vocationally oriented education.

As a final example of the kinds of activities which are helping to extend the use of microcomputers in the schools, we have been assisting in defining some systematic approaches to developing curriculum in vocational educational areas.

Gaps to Fill

There is significant activity in using microcomputers in vocational education. However, microcomputers are not as common in vocational education as they are in the lives of the students. Outside the classroom, amongst the arcade games of PAC MAN and SPACE INVADERS, high school students are in daily combat with microcomputers. Inside the school, only the rare student is learning with or about microcomputers. This problem has two causes.

The first cause is the paucity of systematic curriculum. This is a common lament. We all expect to bemoan the lack of curriculum until we wake up one day with computer based and related curriculum all around us. However, the problem will not be solved until major publishers distribute computer materials as an integral part of their offerings. The tentative, growing efforts of existing publishers to do this are encouraging. But publishers need material. They do not write text books themselves. Publishers produce, market and support material they receive from the creative talent in the field. The long term solution to the limited amount of systematic curriculum for vocational education will be when the creative talent in the field turns out computer literate material and submits it to the increasingly interested publishers.

The second main cause of limited use of computer sophisticated material in vocational education is the computer illiteracy of those involved in vocational education. It is not our fault. We were trained in the 50's and 60's to teach in the 80's for life in the year 2000. While these generational problems have always been with us, seldom has technology moved so rapidly and had such a profound impact on the fabric of our society. The solution is clearly more computer literate vocational education professionals.

Filling the gaps in vocational education, therefore, depends on both the teachers and the creative talent producing publishable material to become computer literate. Raising the computer literacy of the teacher, who is often an author of vocational education material, will soon fill the gaps. At a time when computer literacy for students is zealously advanced, it is possible that the computer literacy of their teachers is more important and possesses more potential impact.

Vocational education in the age of the micro must evolve rapidly. With any luck, this evolution will be natural, as alert teachers become aware of the changing world in which their students will work. Offices, retail stores, and factories will need employees who can work with computers.

Software Sharing in Florida Community College Burroughs Users Consortium

This discussion will be a review of a very successful software sharing experience among several community colleges in the State of Florida. Through this review, I hope to provide some guidelines for others to follow because I believe there are many educational institutions which now find themselves faced with increasing needs for automated reporting and management information but possessing badly outdated computer capability both in the hardware and software areas. Sharing of hardware resources is an obvious option; however, I believe sharing in existing software and software development projects can provide tremendous benefits.

As in the case of so many successful efforts, a number of factors came together to give this consortium effort for software sharing an auspicious beginning.

In the Spring of 1977, Pensacola Junior College (PJC) had recently purchased a new Burroughs computer system with greatly increased capability. All of the software was woefully outdated and card oriented. The major users had high expectations for on-line terminal support to such diverse areas as Student Records, Admissions, Registration, Business Office, Personnel/Payroll, Purchasing, and Accounting. A reorganization of the data processing function had also taken place and a new Head of Software Development had been appointed. Faced with the need to effect dramatic improvement in EDP support and wanting to avoid reinventing the wheel wherever possible, he initiated a survey or available software for his Burroughs computer with on-line registration his priority. His calls throughout the State turned up others who were faced with the same problems and some exemplary software which had potential for adoption at PJC. Daytona Beach Community College (DBCC) was then in the process of acquiring a new large scale Burroughs computer and had embarked on steps to modify software obtained from Polk Community College (PCC) to run on the new computer.

At the same time, there was already a great deal of interest in DBMS. Both PJC and DBCC were committed in that direction and the PCC software was being modified for operation with DMSII, the Burroughs DBMS which was highly regarded by the industry at the time.

During fiscal year 1976-77, the Florida Department of Education adopted a more direct approach to promoting effectiveness, efficiency and equity in educational computing. The 1977 Florida Legislative funded the first year (77-78) of a three year Educational Computing Project. The purpose of the project was to continue to improve effectiveness, efficiency and equity of educational computing in the State of Florida. Improvements would be recognized as new or improved services which met specified goals, reduced unit costs for computing services, and the availability of those services deemed necessary for effective education in all districts and institutions. Emphasis would be on the acquisition of common applications and the promotion of the sharing of common applications between school districts, community colleges, and universities. Monies were provided for a project staff and "seed money" grants to support sharing efforts between institutions in the State.

Before the legislative action had been completed, representatives at the Florida DOE Division of

Community Colleges began encouraging PJC interest in sharing software in the early Spring of 1977. This interest led to a meeting between the PJC Director of Management Systems and the EDP Director at DBCC while at the Spring FAEDS Conference. This meeting was also attended by representatives from St. Petersburg Junior College (SPJC), the Division of Community Colleges and the future Director of the Florida Educational Computing Project (FECP). There was immediate agreement between PJC and DBCC for the sharing of the on-line Registration software which was being converted to run under DMSII. DOE representatives urged the expansion of the sharing group to include all of the community college Burroughs computer users--Polk Community College (PCC) Brevard Community College (BCC) and Pasco-Hernando Community College (PHCC). Subsequent meetings of this group, then known as the Burroughs Users Consortium, led to plans for sharing between the various colleges, offering support to other schools, and a common data base effort.

Grant money provided by DOE as "seed money" to those efforts was never very high, but was sufficient to engender the spirit of cooperation between the various institutions and to "encourage" executive management at each of the participating colleges to provide necessary support resources to the planned projects. In the case of PJC, the $10,000 first year grant was coupled with an agreement by Burroughs to lend software and equipment for testing the DBMS and on-line registration. This support persuaded PJC executives to commit to long range support of the project which included $50,000 in programmer/analyst development time and nearly $150,000 in additional equipment and systems software costs.

During the four years PJC received grant monies of approximately $100,000 under the Burroughs Users Consortium - an additional $31,000 was received for another consortium effort in Personnel/Payroll - for the Student Record (Admissions / Registration) and Financial Aid Projects. Benefits accrued at PJC amounted to approximately $500,000 in areas of cost avoidance in not redeveloping available software, savings in processing time, savings in administrative time or student time, savings in supplies and even savings due to reorganization and decreased man hours required to perform certain functions. Several years were also saved in just getting to the position of having the needed high quality software. Benefits for the consortium Financial Aid project effort are approximately $334,000 for the 81/82 year.

The Burroughs Users Consortium received funds for four years; however, the State is no longer providing funding for such projects. Although the tempo of sharing activity can be expected to decrease significantly, the participating colleges intend to continue a cooperative relationship.

While initially there was certainly interest in tapping the newly available state monies, this group hung together longer, and produced more measurable and effective results than other consortium efforts. The group went on to a shared development effort in producing a complete Financial Aid Support System which is regarded as the best in the State. The reputation of the group encouraged additional colleges to purchase Burroughs computers and adopt the Student Record, Admissions, Registration, Financial Aid software and data base systems used by the nucleous group.

Seminole Community College (SCC) Gulf Coast Community College (GCCC) and

Lake Sumter Community College (LSCC) all have recently purchased Burroughs computers and have taken steps to adopt the Consortium software.

What factors led to the success of this particular group in their Consortium efforts? I believe there is a multifaceted answer to the above question.

Certainly timing was an important influence. The perceived need, enthusiastic participants and state support all seemed to come together in a very auspicious manner for this group. However, other group efforts in the state didn't seem to realize the same results although they may have had the same beginning needs and support.

The State monies were never abundant, yet with good management were sufficient to elicit a number of successful cooperative efforts.

The management within the various DOE Divisions--the FECP and the Division of Community Colleges--did a superb job of helping fledgling cooperative efforts with advice and grant allocations which encouraged those projects which had the potential to achieve the most in terms of providing equitable computing to education. Sometimes we project participants were frustrated that the grants received didn't go further to cover our needs; however, in retrospect, I believe they did a great job with the resources available. Their administrative rules and procedures required complete project proposals when making application for grant monies. The participants were then contracted to perform/achieve the proposed goals. Progress reports were required to ensure compliance. There certainly were some areas of dissappointment when some other groups split up or failed to meet their proposal goals, but the administrative procedures undoubtedly did much to ensure some degree of success in the various projects.

The Burroughs Users Consortium participants had a commonality which accounted to a large degree for the cohesiveness necessary to a successful effort. Most of the participating colleges were committed to data base--DMSII-- and were using or planning to use the same general Student Record--Admissions, Registration, Financial Aid--software system. This made communication in the group better and virtually guaranteed that improvements or developments would be shared and implemented by others in the Consortium.

The most important factors, I believe, were general group dynamics. For success in any group effort, there must be those who are willing to do more than their share, those who will shepherd less committed members of the group, and a recognization that the success of the group project is more important than any individual success. Coupling these factors with enthusiastic leadership and a belief that the goals were achievable and worth going after led to the realization of concrete achievement through the consortium effort and a commitment to continue the relationship.

Even after four years of intensive effort by DOE and others there are still institutions within the State of Florida desperately in need of assistance if they are to realize the degree of EDP support needed today in education. We still have colleges with antiquated computer equipment and badly out dated software. Some school districts have no EDP support at all. I'm sure other states find like situations within their own boundaries.

I am an enthusiastic supporter of further consortium efforts such as described above and believe a modest amount of State administered monies can promote tremendous achievement toward more equitable EDP support to education.

I would make the following recommendations for any future action within the State of Florida or other States:

1. Develop & promulgate clear set of goals with established priority orientation to specific objective areas such as student records, financial aid, automated reporting, or management information systems.
2. Establish central staff for providing assistance to fledgling efforts, and administering the various projects and grant apportionments.
3. Insist on committment or matching investment by participating institutions.
4. Insofar as practicable, the consortium efforts should be made up of membership including both the special objective areas such as financial aid and data processing specialists. There should be infused within such groups enthusiasm for the project and a commitment to the whole rather than the so often encountered "whats in it for me" attitude. There need to be participants willing to go that extra step and provide cohesiveness for those less enthusiastic participants.

- A.R. Monish

AN ON-LINE LIBRARY CIRCULATION SYSTEM

Billy Pennington
Head of Circulation
Mervyn H. Sterne Library
University of Alabama in Birmingham

In order to adequately serve their users, libraries must be able to quickly access and process information. As the volume of collecting, organizing and disseminating information to patrons has grown, libraries have looked to alternative ways of accomplishing these functions. The application of automation has been one of the most promising ways of coping with the phenomenal expansion of knowledge and the increasing demand for information.

Mervyn H. Sterne Library at the University of Alabama in Birmingham is utilizing automation to speed its operations and to improve services to users. In the past three years, the Library has developed fully automated circulation and acquisition systems, or rather subsystems, since these are only two components of a planned multifaceted automated library system.

Operating on the main frame at the University's Central Computing Center to which the Library is linked by terminals, the on-line systems are running smoothly and successfully. Service to patrons has improved and operation of the circulation and acquisitions functions is faster, more efficient and more accurate than when performed manually. The staff of the computer center has worked closely with the library staff since inception of the project. This excellent working relationship has contributed in large part to the success of the project. Along the way there was continuous interface between our staff and that of the computer center.

Background: UAB is a young and rapidly growing institution. In 1970, the Library had fewer than 20,000 volumes. Today we are approaching 550,000 volumes. In 1981, the Library circulated more than 217,000 items to some 12,000 patrons from UAB and Birmingham. Rapid growth of the collection and of the patron population served without any significant increases in staff created many problems. Control of the collection and of the circulation function was tenuous at best.

Early on, the library administration recognized the value of automation in dealing with the library operations, especially in such a rapidly growing unit as ours. In 1973, a batch fund accounting system was developed. In the next few years this system grew and was expanded. By 1977, the library administration recognized that a more comprehensive system was needed. The need for circulation relief was also evident.

The original idea of the library administrative staff was to purchase a minicomputer to develop a circulation system. The University Computer Center, however, had veto power of all computer hardware purchases and shot down the idea with the justification that they (the center) could provide the service more economically and efficiently. There was some initial skepticism, but discussions between the library staff and the computer center staff were opened. And in early 1978, a contract was signed with the computer center to design and develop the acquisition and circulation sub-systems. The acquisition sub-system went on-line in early 1979, and the circulation system went on-line in June, 1980.

Acquisition System: The on-line system was an elaboration of the batch system developed earlier. Basically the system provides full tracking of an item from the time it is selected, i.e. the decision is made to buy, until the item is cataloged, at which time it is transferred into the circulation system. Once an item is selected for purchase, it is entered into the item data base; if funds are available, a purchase order is generated. If funds are not available in the account to be charged, the title remains in the item data base until such time as funds are made available. This enables the encumbrance of funds on short notice. Once purchase orders are generated, titles are moved to the order data base and account balances are adjusted automatically. When materials and invoices arrive, they are worked at the terminals and payment is authorized. The system interfaces with the University business office and authorizes the payment of invoices. In addition to purchase orders, claims and cancellation notices, notification slips to university departments and other reports are generated by the system. Funds are allocated in the system and tracked by school, department and fund/accounts and types of materials.

Circulation System: The circulation system is a full inventory system that can, at any point in time, identify the status of any item in the collec tion, whether it is on the shelf, checked out, at the bindery, on reserve or missing or lost. The system maintains a current file of all patrons using the Library, and it provides a record of all current transactions of patrons on-line, as well as completed transactions. This historical record is purged to fiche on a quarterly basis The system provides access to any title or volume in the collection, either by its unique ID number or by call number, by OCLC number, by author or eventually by title. The system places materials on reserve for faculty and places holds on items requested by patrons. And finally it generates overdue notices and a variety of statistical reports.

Equipment and Operating Software: The main frame computer located at the central computer center is an IBM 4341 Group II. The terminals are IBM 3278 linked to the computer via leased telephone lines. Three terminals are located in the Circulation Department, three in the Acquisitions Department, one in the Reference Department, and one in the Cataloging Department. This latter terminal may be relocated as the conversion effort is completed. The terminals at the circulation desk are equipped with OCR wands (Optical Character Recognition). OCR labels were utilized for labelling the books and patron ID cards. There is also a printer which is accessible to both Acquisitions and Circulation. The operating communication software is IMS-DC. All application programs are written in COBOL.

Conversion: One of the largest parts of any automation project involving a large number of records is the inputting of those records into the data base. This can be done manually or, if the records are in machine readable form, they can be manipulated to the format of the data base.

Fortunately for the Library's operation, almost all of the library's holdings had been previously converted to a machine readable format and were stored on tape. These tapes were utilized to build the data base for the circulation sub-system much more quickly than if the records had required inputting manually.

Future Plans: The Library is not resting in its automation activities. The title search capability will be added to the circulation sub-system within the next six months. This will essentially make the system a keyword title index. We expect to begin development of serials check-in and audio-visual book components as soon as funds are available. An on-line catalog with subject search and user interactive terminals are further down the road.

The path that Sterne Library followed in automating its operations has been an exciting one. We are coping with increased and more complicated operations in a rapidly growing institution. We are utilizing our staff more efficiently. And we have gained control of a ballooning annual circulation. But most importantly, we have improved the level of service to our patrons.

THE COMPUTER: AN AFFECTIVE MACHINE

by Jeff Poser

During the last few years, the microcomputer has proven itself a useful instrument in stimulating cognitive development of students. However, the microcomputer's capabilities need not be limited to increasing student scores on tests. At Valparaiso Elementary School in Okaloosa County, Florida, an Apple II is being utilized as a device for improving behavior. This computer has become a significant factor for modifying behavior in the Emotionally Handicapped class. What was formerly seen as a "teaching machine" to aid growth in the cognitive domain, is now a valuable teacher tool for shaping growth in the affective (or behavioral) domain.

With the acquisition of the Apple II microcomputer, our Emotionally Handicapped classroom became the laboratory for experiment and adaptation of the computer to the needs of a group of special students. The term'Emotionally Handicapped', refers to those children who, due to inability to adapt socially and emotionally, cannot function in a regular classroom. These children display a wide range of behavioral patterns. It is the goal of the E.H. teacher to isolate these behaviors and attempt to eliminate them. The approach used involves various strategies of behavior management. Motivation for social cooperation is provided and positive emotional responses are rewarded. The child, gradually assumes new, more appropriate behaviors and is eventually prepared to re enter the mainstream. Once the teacher has targeted desired behaviors, the task is to provide effective motivation and reward. Students in such a program show dissimiliar abilities in age, intellectual achievement and emotional maturity.

In this particular classroom they ranged in age from 5 to 12 and showed I.Q.s ranging from 68 to 140+. Behavioral patterns included unsocialized and socialized agressive children, as well as those that were over inhibited. Generally, these students possessed poor self-images and were unable to achieve academically and interact socially in a regular setting. In a class with such a wide range of academic abilities and a wider range of personality differences it is difficult to find motivators. The Apple II was a teaching aid with common appeal. Its use could be adapted to meet the academic needs of various learners and they would change their behavior for the opportunity to work with it. For such a diverse program, the Apple II became an invaluable instrument for motivating behavioral change.

The classroom for the Emotionally Handicapped employs behavioral management techniques, so it became apparent that the procedures for using the microcomputer would follow along such guidelines. When the E.H. class acquired the Apple in January of 1981, both teachers and students became familiar with its use by just "playing" with it. No one had much experience with computers, so everyone learned by trial and error. After playing with the computer for over a month it became evident that its use was highly motivating to the students. Knowing that in order to elicit a desired response a motivational component is needed, the use of the computer fit well into the context of a behavior management program. At first the student could use the computer at the end of the school day if he had earned

Free Time. Free Time was earned by the students for displaying appropriate behavior during the school day. If the child stayed out of Time Out I (study carrel in an isolated part of the room), he earned Free Time. A student would be sent to Time Out 1 if he received three of the same type of "FINE" for inappropriate behavior. The "FINES" were determined by the students at the beginning of the year and included such behaviors as "OFF TASK" (not following directions, refusal to comply), "BOTHERING OTHERS" (hitting, kicking, throwing objects), "INAPPROPRIATE NOISE"(yelling, talking at an inappropriate time), and "OUT OF SEAT".

In order to maintain a high level of motivation, disks with games were acquired. The games not only developed visual motor coordination, but also provided growth in social skills. At the end of a day, a number of children would earn"Free Time". This produced a social situation in which each individual would have to rely on his social skills for interacting with a group. If the motivation to use the computer was high enough, then the student would have to share with the others. Also the group as a whole would have to develop their own organization for using the computer. (The number of turns one gets and the game to be chosen were problems that the group confronted each day.) The development of the group using the computer was left entirely up to the students. If a student did not cooperate with the group as a whole, then an individual could use the court system, a system designed to let the child and his peers have some control over their own behavior. When a child does something that another child thinks is wrong, then he sues that child and is judged by his peers. If the child is found guilty, then a student who has shown the best behavior from the previous week prescribes a punishment that is appropriate for the action. When the child is able to have some control over his social development he is more apt to internalize it. The motivation afforded by the computer makes it possible for the child to work in those group situations and enhances social development.

Besides using the computer as a stimulus to organize social interaction and organization, a group activity using the computer was also established. During group time, disks which provided concepts concerning home safety, visual perception, language, and math were put into operation. This also made it possible for growth in social development in a more structured setting. It gave those students that were unable to gain "Free Time" an opportunity to engage in a social development activity with a high motivational factor.

Later, it became evident that the computer was not being utilized to its fullest potential. there were 330 minutes that the student was in the classroom and the computer was only being used for 20 minutes of "Free Time" and 30 minutes of group activities. At this time a variable interval reinforcement strategy was adopted. With the computer being the motivation, the desired behavior became "ON TASK". This means that a minute timer is set to a variable schedule. The bell is set to either ring in one minute or in ten minutes, but student does not know what interval is set. When the bell rings and the student is displaying "ON TASK" behavior, he has the opportunity to use the computer. The teacher will flip a coin to determine if it is an assignment programmed by the teacher, or a free choice by the student. If the student calls the right side of the coin he may play a game or do an assignment. If he chooses an assignment he will earn additional points in the classroom. If more than one student is "ON TASK" when the bell rings, then those students that are "ON TASK" will be able to select a piece of paper from the computer box. If they select the paper marked "COMPUTER", then they are able to use the computer and the teacher flips the coin. This strategy not only promotes the desired behavior but also enables the student to deal with frustration. When the student picks a blank paper from the computer box, he has to learn how to cope with the feeling of frustration. In coping with this emotion he has taken another step towards social maturity.

The results of using the computer for changing behavior have been very satisfying. Students that demonstrated behavior patterns of being passive aggressive, with above normal intellect, virtually decreased their need to use Time Out to nothing. With students that showed aggressive, unsocialized behavior with normal intelligence, a 50% decrease in use of Time Out was noted. However, students that showed hyperactivity and learning disabilities, did not show any reduction in "OFF TASK" behavior or use of Time Out. Like many strategies used in working with the affective domain of children, the use of the computer was effective in changing behavior of only certain students.

Besides using the microcomputer as a motivational device for working with students, it has produced a more effective program for changing behaviors because it enables the teacher to keep more efficient records on the students. In order to note change in behavior, it is vital to keep as much data as possible describing each individual's behavior. Prior to the acquisition of the computer, the teacher could only record specific behaviors for each student on a tally sheet. The behavior charted would depend upon each individual's specific problem. At the end of the day, the total would be added and transferred to a graph. This charting activity would occupy a great deal of time at the end of the day in order to compile all necessary information. Now, using the APPLE II, the teacher is able to keep specific behaviors and additional information that is pertinent to the child's development. Besides storing the number of times the student displays a specific behavior, the computer keeps daily anecdotal reports, the number of times a student uses Time Out, and the daily grades each student receives in each academic subject. The disk that is used will sort, add, and print out only the information that is needed.

When the end of the grading period arrives, the teacher can determine whether the child has shown an increase or decrease in specific target behaviors. This is still graphed, however, so a visual representation of the child's behavior is seen to demonstrate a correlation between behavior and dates. The storing of the number of times a student used Time Out also enables the teacher to recognize any increase or decrease in its use. Also, with daily grades already recorded in the computer, the task of filling out academic report cards is simplified. The most beneficial storage of data is the accumulation of daily anecdotal comments on each student, which enable teacher to observe patterns in a child's behavior. Also, it provides insight into the child's perception of his world and relationships. With this information, the teacher can become more objective and prescribe a better program for the student's affective development.

Not only has the microcomputer been an asset in prescription of behavioral strategies, but it has produced a better means of maintaining the many forms and paperwork used in developing an individualized behavioral program. Before using the computer all forms used for daily behavioral observation were typed or handwritten to suit the needs of each individual. In the E.H. class, there is a high rate of transition. Whenever a new child enters the program all the composite forms have to be altered in order to include that individual. Now all the forms are stored on disk. When a new student enters, we can easily alter the form that is needed. By storing forms on disk, the teacher can also go back and improve upon them, rather than constantly make new ones. The use of the computer frees the teacher for more time in planning a better program for each child.

By using the microcomputer in the E.H. classroom, a more productive environment for altering behavior has been established. The computer has proven to be an affective machine by providing the motivational reinforcement needed to shape behavior disordered children in our over stimulated world. It has also been used to develop social skills in group situations and to program instruction for these skills. Besides working directly with the student in modifying his behavior it has given the teacher the opportunity to produce a better program for children with behavioral problems. Perhaps, in the near future, the computer will be useful to all teachers in aiding the overall growth and development of children.

COMPUTERS+BOOKS+KIDS=REAL LEARNING

Elizabeth S. Wall

Media Consultant

In 1925 an American inventor who was so enthusiastic about one of his inventions as an aid to learning, predicted that this invention would make books obsolete in 10 years. The man was Thomas A. Edison and the invention was the motion picture projector. Well, Edison's prediction didn't materialize for the motion picture projector, nor did any other succeeding predictions about tape recorders, video discs, or television sets replacing books. Books are alive and well living side by side in schools with all these machines.

Today we're dealing with computers and books in the learning equation because we know that computers aren't going to obsolete books either. First let's see how important the new machines are and then how books help in understanding them better. In August of 1981 the U.S. Dept. of Education stated that about one-half of the nation's school districts use a total of 52,000 computers for instructional purposes, mostly "computer literacy." Microcomputers now outnumber timesharing terminals by 3 to 2. Just over 7600 schools have computers. These numbers and any others we get from microcomputer manufacturers all tell us that the computer is rapidly becoming commonplace in school. All of us, educators, teachers, librarians, principals, superintendents and parents must look for materials to make ourselves and our kids computer literate.

To accomplish computer literacy, the machine and its software are the critical ingredients and of course must be carefully chosen to reflect the needs of the curriculum. That acquisition process is extremely important and involves a whole set of unique criteria which is being dealt with in many other meetings in this conference and others. Therefore, in this meeting, we'll only discuss how computers plus books can be a part of a successful computer literacy program for kids and, incidently, the adults working with them. We'll look at current computer books and explore how they can relate to kids and machines to produce the "real learning" which I define as meaningful, efficient and progressive learning, not a one shot deal.

Visualize with me the situation at Garden Elementary School in Venice, Florida where one microcomputer will arrive for 956 children and about 45 teachers. How will books introduce these users to the machine, prepare them to use it, and help them become independent and creative with it?

First, books introduce the computer by showing kids what one looks like, describing what it is and how it's used; second, books prepare children to use and program the machine, by explaining the keyboard, monitor and other input peripherals, and showing how different parts work to make the computer do what the user wants it to; and third, computer reference books will help kids become independent learners.

Let's begin with introduction, where books will demystify the computer. Garden School like many schools face the plight of beginning a computer literacy program for teachers and students, many of whom have no prior experience with computers. The teachers at best are just disinterested, others are suspicious, afraid or even antagonistic about the arrival of the computer.

To meet these attitudinal problems, you'll find books present a convenient, pleasant, and relaxed way to "meet the computer". The printed word dispells the myths and opens the way for a more friendly experience with the computer. While everyone is waiting for the computer to arrive or for the "hands on" sessions to begin, a classroom or library learning center containing books about computers and their role in our lives is an inexpensive and nonthreatening way to allay fears and get kids and teachers turned on to computers before the machines arrive in school.

Books to use in this initial "before you sit down at the machine" preparation time are called awareness books or first books. They tell about the computer and are useful without a machine. They're written to help users discover that computers are neither inscrutable nor magic, just a part of everyday life working for us and the community. There are several good computer awareness books available now for children and I'll tell you about them later.

Fortunately, however, most children in the elementary and middle schools are eager, enthusiastic and ready to start right in on the computer. But wait, herein lies the next problem for Garden School, limited time per child on a complex machine. A machine that must be carefully loaded with a program and which demands precise instructions sent to it via a nonalphabet keyboard which is not at all familiar to most children. Many of us here have already experienced the fact that it takes a child an interminably long time to type his name or any instructions on the keyboard. The impact of dealing with the keyboard can be alleviated with computer literacy books, those which are used with the machine. Some of these are programming texts and kids will be better prepared to use their limited hands-on time using them. We'll look at some recently published children's computer literacy books later.

Finally, the computer book as a reference. I can remember one Friday afternoon when I first learned to use a word processor program, APPLEWRITER. I wrote up a storm that weekend, going to my computer- expert husband everytime I needed to do something different. Panic hit me when he prepared to leave Monday morning on a trip. "What will I do," I asked, "if I get stuck?" Very reasonably he replied, "As long as you can read, you ought to be able to do anything!" And he handed me the instruction manual and left. With that assurance, and a lot of trial and error on my own I was able to be independent.

We're fortunate today in finding good computer books for kids. As little as 4 years ago most books about computers were written by and for an adult computer science major or by and for the "computer freak" hobbyist. Forunately this has changed and we now have computer books written by authors who are educators and children's book writers who are working with kids and computers.

We know that one book will not solve the three problems we predicted for Garden School's computer literacy program, so let's look at several examples of computer awareness (no machine needed) books. For the young child, THE COMPUTER ALPHABET BOOK by Elizabeth S. Wall (Bayshore Books, grades 2-5) uses the familiar alphabet to introduce and explain common computer terms. A child's first grasp of a new concept is with random words which he then sequences and relates to familiar experiences. This first book takes a group of critical computer words and defines each in the language and experience of primary age children. At the end of the book the complete novice will know a good bit more about computers. KATIE AND THE COMPUTER, by Fred D'Ignazio and Stan Gillian (Creative Computing grades 3-6) is a colorful, fast action adventure story which fantasizes how a computer works inside. Using the

Alice-Through-the-Looking-Glass technique, the author takes Katie on a fantasy journey inside the computer to the land of Cyberria where she learns about the processing of a command from input to output, but only after evading the Monster Bug with his bubble gum lasso. COMPUTERS IN YOUR LIFE by Melvin Berger (Crowell, grades 5-8, illustrated with photographs), HOME COMPUTERS: the Simple Facts by Scott Corbet (Little Brown, grades 4-8) and LIVING WITH COMPUTERS by Barry Blakeley and Robert Lewis (Crane-Russak Co. grades 5-8) reveal the work of computers at home and in the community. EXPLORING WITH COMPUTERS by Gary Bitter (Julian Messner, grades 3-5) is an awareness book which intends to be an introduction to the world of computers. It contains short sections on hardware and software, computer types, uses, misuses and careers. There are a few computer related activities in the back which can be worked without a computer in hand. THE STORY OF COMPUTERS by Donald Spencer (Camelot Publishing Co., Grades 4-9) introduces children via a humorous, simple narrative, to what computers are, who invented them, different kinds of computers, and how one works. Using cartoon-like drawings, the author gives examples of input and output devices and simple programming.

Some examples of recently published literacy books which begin in simple ways and help teachers and students discover how to "talk to" a computer include the following: BE A COMPUTER LITERATE by Marion J. Ball and Sylvia Charp (Creative Computing, grades 4-8) reveals a basic understanding of computers through identifying and explaining computer words and terms. This colorful, appealing paperback contains chapters on different kinds of computers, how to communicate with the computer by understanding a language and how to write a simple program. The final chapter tells how computers work for us. ONE COMPUTER, 30 KIDS by Kay Richardson (Deka Press, grades 3-6) is a 3-ring notebook containing basic student worksheets to help students with the mechanics of BASIC programming before, during and after their "hands-on time". The exercises are grouped by BASIC statements. A teachers' guide is included. Teachers do not need a machine to utilize this book. COMPUTERS FOR KIDS by Sally Greenwood Larson (Creative Computing, grades 3-8) contains information for very young programmers. It has large type and is easy to read. The author makes it easy to find out how to get your computer to do what you want it to do. There are editions for the APPLE, ATARI, and TRS 80. A fast paced activity book which spells out many ways to enjoy a new personal computer is THE CREATIVE KIDS GUIDE TO HOME COMPUTERS by Fred D'Ignazio (Doubleday, grades 6-adult). COMPUTER LITERACY - Problem Solving with Computers by Carin E. Horn and James L. Poirot (Sterling Swift, grades 10-adult) is a well organized textbook discussing computers in both technical and nontechnical terms for parents and teachers. COMPUTERS FOR EVERYBODY by Jerry Willis and Merl K. Miller (dilithium Press, grades 10-adult) provides readers, with enough information about computers to select one that is appropriate to their needs. How's that for openers?

Finally, WHAT CAN SHE BE? A COMPUTER SCIENTIST by Esther and Gloria Goldreich (Lothrop, Lee and Shepard grades 3-6) takes the reader along with Linda Wong, a computer scientist as she studies problems with clients and sees how computers can help solve their problems. Text and photos reveal a fascinating introduction to the work world of computers and the variety of job opportunities there.

Now you know about some books, currently on the market, which will certainly live happily in schools side by side with computers and all the other machines. And as we look again at our original equation with all the variables we can interchange with computers, i.e. motion picture projectors, tape recorders, record players, or video disks, we can see that the constants are still books and kids and I for one don't think that will change for a long while.

RESOURCES

Dateline:Tomorrow, CREATIVE COMPUTING, Morristown, NJ 07960, July 1981

Library Update by Barbarie Harvie, COMPUTERTOWN USA! News Bulletin, Menlo Park , CA 94025, January 1982

Computer Literacy: Finding Effective Resources by Barbara Kurshan, Recreational Computing , Menlo Park, CA 94025, May-June 1981

Bibliography

Ball, Marion J. & Charp, Sylvia. BE A COMPUTER LITERATE. Morristown, NJ: Creative Computing,1978

Berger, Melvin. COMPUTERS IN YOUR LIFE. New York, NY: Crowell, 1981

Bitter, Gary. EXPLORING WITH COMPUTERS. New York, NY: Julian Messner, 1981.

Blakely, Barry & Robert Lewis. LIVING WITH COMPUTERS. New York, NY, Crane Russak Co., 1978

Corbet, Scott. HOME COMPUTERS. Boston, MA: Little Brown, 1980

D'Ignazio, Fred & Gilliam, Stan. KATIE AND THE COMPUTER. Morristown, NJ: Creative Computing, 1979
..............CREATIVE KIDS GUIDE TO HOME COMPUTERS. New York, NY: Doubleday, 1978

Goldreich, Gloria. WHAT CAN SHE BE? A COMPUTER SCIENTIST. New York, NY: Lothrop, Lee & Shepard, 1979

Horn, Karen E. & Poirot, James L. COMPUTER LITERACY. Austin, Texas, Sterling Swift, 1981

Larsen, Sally G. COMPUTERS FOR KIDS. Morristown, NJ: Creative Computing, 1981

Richardson, Kay. ONE COMPUTER, 30 KIDS. Indianapolis, IN: Meka Press, 1980

Spencer, Donald D. STORY OF COMPUTERS. Ormond Beach, FL: Camelot Press, 1977
...................COMPUTER DICTIONARY FOR EVERYONE. New York, NY: Scribners, 1979

Wall, Elizabeth S. COMPUTER ALPHABET BOOK. Nokomis, FL: Bayshore Books, 1979

"COMPUTER-MORPHOSIS"

by
Sister Catherine Weaver, O.S.F.

Growth in a cocoon yielding a beautiful butterfly is a morphosis. A change in form is its essence. Thus, I thought it an excellent way to explain what takes place when a decision is made to grow and change form. All morphoses have a beginning and this one has its origin in 1973. At Lancaster Catholic High School in Pennsylvania during that year, I was Mathematics Department Chairperson. I had completed studies for a Master's Degree and was exposed to Computer Programming on a "grandaddy Wang". It was the "grandaddy of all grandaddies." Two days after it was installed, and before we had an opportunity to use it, the central unit was struck by lightening during a thunderstorm. We waited while experts resorted to time-honored methods trying to create an operable system again. The experience of working on tape with punched holes, waiting in line to use the terminal (of which there were only two), spending hours to find errors and being apprehensive over whether a program would run after tedious use of Basic language did not leave one with a good first impression of computers. Undaunted as all Mathematicians are by difficulties, however, I proceeded to evaluate and decide how best we could get a smaller system at the high school with packaged software. The latter was an absolute necessity because only one other teacher in the department had computer experience and his background was limited to punched cards on a Monroe System.

Many companies were contacted, many salesmen were invited in to demonstrate their wares, many large systems were explained. The prices were quoted. Finally, the question raised was the software available for Algebra through to Calculus. Each time, the answer was in the negative. Yet, we were encouraged to buy the system. It was very frustrating as I was unable to find any small computer on the market below $5,000, and with the associated programs needed. For five years, I tried any and all avenues open to me, contacting companies from Maine to California where I heard microcomputers were then available. By 1978, when I left the Lancaster area for my present position, microcomputers were available but the prices were still beyond our budget. Very few and limited programs were available for mathematical needs.

In my new position, as Principal of a secondary school with over one thousand students, I was extremely busy. I wasn't able to concentrate on just that one need, but had to develop many areas within the school community. I set up a five year plan for the school indicating that in the third year, everyone would update curriculum. There it was again, "staring me straight in the face". We had a qualified teacher for Computer Programming and the PET Commodore Company was offering five terminals for the price of three.

In February, 1980, I approached Ms. Anne Wilson, who taught Mathematics at our high school. I mentioned to her that I had checked her transcript file and noticed that she had six undergraduate courses in Computer Programming. I inquired as to how she would feel about teaching the subject. I was pleased to hear her reply, "I'd look on it as a challenge." So, I made final arrangements to acquire the five terminals, elicited the husband of one of our faculty members to give a six week course. He taught Computer Programming at a nearby college and came after school to begin some basic instructions with ten of our faculty who indicated they were interested. After two or three weeks, Ms. Wilson was the only one left who still attended the course.

Recently, I spoke to Ms. Wilson, now Mrs. Sabol, to garner from her experiences the growth that has occurred in our program. She told of how she worked and played during her Service Period each day following basic

488

instructions in the Manuals. She reminded me of a National Council of Teachers' of Math Convention that she attended in March of that same year and where she did more "hands-on". She procurred many free materials and games to use. The text, "Basic, Basic" by James Coan, which she got at that time, will now be used for our Computer II classes next year. Anne proceeded to build background materials and ordered several publications such as: "Byte", "Creative Computing", "Compute" and "Computing Teacher". "Byte" and "Compute" were curtailed eventually as the former was more for those in Business and "Compute" was for home operators. The other publications we still get, read from cover to cover and use constantly to update ourselves and our software. During the summer of 1980, Anne continued to work and play on the PET Commodores, doing Basic Microcomputer Games by Ahl and converting sixty character programs from other sources to use on our forty character PET terminals. I asked her if she was ever discouraged or frustrated when programs didn't work. She replied that frustration came but rather that determining a solution irritated her till she corrected the mistake.

The first course of studies Anne prepared developed around a skeletal outline given to her at a convention. She gave it flesh.

I reminded her, intermittently, to take her time, to feel comfortable with the machines and not to push herself too much the first year.

She told me, later, how she used graphics with print statements initially with the students. Games were also used, namely, Maxit, Battleship and Black Jack. Interestingly enough, she found that the students retained interest only in Maxit, which had definite mathematical implications to a 10 x 10 matrix, positive and negative numbers and definite planned strategy. Teaching these, terminology, a general short history of computers, base 2 and coding form the basics in her first year course. She develops If....then statements, For....next loops and educates the students in using disk drives to record their programs. Fifty-six students in four groups grew to seventy-eight students the following year. At this time, Mrs. Sabol has begun to train a fellow mathematics teacher to take over the first year courses as she begins a second year sequence to the program. We now have one hundred eighteen students registered for both first and second year courses beginning September, 1982.

Speaking to Anne again so as to evaluate what has transpired in the program thus far, she shared with me decisions to delete some of the games and demonstration work used during the first year. She intends to replace them with more "hands-on" practice of a definite mathematical nature. These will run the gamut from Algebra I/ Geometry through to graphing Quadratics. She hopes to develop the second year course to take in Math Analysis and Calculus. In the meantime, Anne worked on a program to use with scheduling homerooms for the administration. She has continued that during the summer months, programming a mailing list for our students.

At this time, I must inject the concept that during our changes, I dubbed the transit "Computer-Phobia" rather than "Computer-Morphosis". Why? Everyone who was encouraged to visit and just play with the machines froze. Comments like, "I have trouble with Math", "I'm scared to try it", raised us only one step above where we'd been before. Some more software was now available and so we dove into "sink or swim" again. Fortunately, the teachers were getting to be less fearful.

Early in the second year, the question arose as to whether we'd done enough or whether we must now branch out into Word Processing for our Business students. Again, the phobia or morphosis, whatever you wish to call it, was overwhelmingly evident as I spoke to the Business Chairperson and a teacher in the department. Fear, frozen, stiffened bodies, negative statements and again the ugly head of Math fear raised itself. Quietly, on the side, the Department Chairperson said, "I'm ashamed to admit it, but I can't do Math". Again, I tried the "dive in or drowned" approach. There they sat, looking sad, those lovely machines that could do so much while the teachers circumvented the room and students peeked in out of curiosity. One of my Vice Principals finally went in to play with one machine a bit each day hoping to pique the curiosity of the faculty, but to no avail. I began to think I was pushing too hard. I examined my conscience almost daily, faulting myself for getting the machines. "It's wasted money", I thought. "They're just not going to use them in the Business Department. Finally, a workshop was offered at a nearby college and I encouraged the two Business teachers to go. With trepidation, they finally agreed, but when I offered to have the school pay so they could get credit, they responded, "No, we'll just take it". Halfway through the course, they announced to me that they decided to pay for the course and get credit, after all. "Finally," I thought, "a glimmer of hope, a hopeful ray is beginning to show". But, again, they retreated into their shell and returned to ask me for a Memory Typewriter in the department, arguing for that against any Word Processors as part of a microcomputer. For months, we debated and to strengthen my own certitude, I returned to undergraduate and graduate classes in computers. Finally, almost frustrated and feeling hopeless, an idea formed to help the entire faculty while giving the computer companies an arena to display their wares. I invited representatives from Apple, Atari and TRS-80 to do "hands-on" with our faculty and Principals from local elementary schools. While still in the planning stages, I brought in Atari with Word Processing software for demonstration purposes. The software prepackaged program made the operational aspects so easy that the teachers spoke up immediately in favor of purchasing five for the price of three. "We're off!", I thought, but many more mountains had to be climbed. The In-Service Day with "hands-on" lit a spark. One teacher purchased a home microcomputer. Several others elected to take a three credit course at a nearby college. One business teacher began to work on the machines with her students. She was proud of the most minute steps toward growth. So, it began. One of our excellent typing students is presently putting our entire student handbook on a floppy disk for later recall and correction through the Word Processor. Several students go to the Word Processors on their own now and work to the joy of the Business teacher, who still keeps a respectful distance.

Then, as I sat composing this presentation, the Science Department Chairperson stopped by to ask what were the chances of her getting some microcomputers for use in the Chemistry and Physics classes next year. My face lit up like the noonday sun as I encouraged her to seek out and evaluate which company had the best software for her academic area.

A second year Computer Programming course is set up and the Office Practice class has produced several projects in Word Processing. The cocoon is opening. A beautiful world is appearing, not just in my eyes, but in the minds of the students. More than this, a new wave of enthusiasm is motivating our faculty members. As they grow, so do the students. A phobia has become a morphosis - from Math - to Business - to Administration - to Science. We're growing! We are now planning to offer Word Processing and Computer Programming for evening courses to adults in our local community with our faculty teaching them. You can see how the worm of an idea through metamorphosis developed into something beyond my own expectations.

I hope all of you will have the same courage to grow and develop in this "Age of Aquarius" with computer technology. The result will far exceed your dreams. The world will evidence new horizons to you - go out to view them in all their beauty.

SCIENCE LITERACY IN THE PUBLIC LIBRARY

Alfred Bork
Stephen Franklin
Ruth Von Blum
David Trowbridge
Educational Technology Center
University of California
Irvine, California 92717
(7]4) 833-69]]

Barry L. Kurtz
Department of Mathematics
San Francisco State University
San Francisco, California 94]32
(4]5) 469-2539

Introduction

This paper discusses a set of highly interactive, graphic, computer based, learning modules developed in the Educational Technology Center at the University of California, Irvine. These modules were developed with support from the Fund for the Improvement of Postsecondary Education. They address public understanding of science, particularly the nature of scientific knowledge.

The primary testing environment has been the public library, although the materials have also been tested in other places. They are intended for a broad general audience, one which may not have had previous experience in using computer based learning material. They are also very suitable for home computer use.

We are discussing the marketing of these units with commercial companies, but no full arrangements have been made. Materials are available on a trial basis for those with suitable computers.

Learning Modules

These materials focus on the nature of scientific knowledge, concentrating on how the scientist goes about developing scientific theories. This perspective is difficult to present with any learning medium. So the issues are seldom discussed with the majority of students in our schools and universities. But if we are to have citizens who understand science, understand it sufficiently to behave intelligently in a society that is heavily influenced by science, "public understanding of science" is important.

The materials are developed, at least initially, for informal learning situations. Initial testing has been in public libraries, drawing on a wide range of individuals. The material has also been tested in other formal and informal learning settings.

The materials do not aim at a specific age group. Rather, they interest a wide range of users, with many different talents and capabilities. They make some adjustment to the talents of the users at the level of individual questions and sometimes in broader areas.

The materials are computer based learning modules, carrying on an interactive conversation with each user. Each unit takes one and a half to two hours, assuming an average user. This time varies considerably, depending on individual skills and interests.

The larger units are broken into ten or fifteen minute segments or modules. The user can easily move around among these individual segments. Each segment is somewhat self-contained, often beginning with a review of earlier material. At times it is suggested that the user return to earlier segments, based on performance in the current segment.

The computer dialogs make extensive use of graphic capabilities in a variety of ways. Thus, they stress visual learning aspects as well as textual learning aspects.

We have not made any careful tests of the average time between user interaction. A group of users recently estimated that about fifteen seconds, on the average, elapses between the time a student finishes typing in a response to a given question and the time when the student is given another question to respond to. This time will vary; sometimes as much as one or two minutes will elapse, while at other times only a second.

The interactions are in English. That is, the user is not asked to respond in restricted fashions. The programs respond correctly to a wide variety of ordinary words, phrases, and sentences. Multiple choice is almost never used, as is characteristic of other material developed at the Educational Technology Center. The units frequently use "pointer" input, where the user points to something on the screen to indicate a choice.

Pedagogical Style

Each of the programs was developed by a separate group of authors, chosen for their skills in aiding students in the learning process, particularly their interactive skills. Thus, no single pedagogical model or style is followed in all the materials, as different authors have different styles. Nevertheless, one aproach is common enough to a number of the modules to deserve mention.

This approach is based on the learning cycle ideas, as initially developed by Robert Karplus at the University of California, Berkeley. Working from a Piagetian basis, Karplus distinguishes between three stages in the learning cycle, an initial experiential stage, in which the student builds up intuition about the activities involved; a second more formal tutorial stage, often thought of as the sole stage in learning; and a third stage in which the student must act with what has been learned.

In our work at Irvine we have extended the learning cycle to allow for internal mastery testing and feedback loops. We refer to this strategy as the mastery learning cycle. The mastery learning cycle puts the student in a situation similar to that in which a scientist would work. Thus, it is appropriate in a project in which the main objective is scientific literacy in a broad, general population. The experiential stage corresponds to the initial gathering of evidence, the empirical investigations on which later theoretical structures can be based. The scientist develops hypotheses, makes predictions based on the hypotheses, verifies these predictions with further empirical evidence, and refines the tentative theoretical conjectures. Students follow this process in our modules.

Thus, a principal pedagogical tactic used in many of the units is to have the student behave like a scientist in investigating some scientific phenomena. The student gathers evidence and uses this evidence in the testing of hypotheses. As with scientists working with "real" theories, we do not expect the student to come up immediately with a reasonable hypothesis. In many modules the student deals with partial hypotheses that prove to be inadequate.

Design Style

The Educational Technology Center has always been concerned with the visual and temporal structure of its computer dialogs. The computer is a new and unique learning medium, with properties different from books, lectures, films, and human tutorial aid. A full realization of its potential demands careful attention to the media characteristics that are important in learning. These issues are amply discussed in other literature available from the Educational Technology Center. We will review the principal factors.

The screen is a new kind of "tablet" on which information can be placed, either by the student user or by the computer program. This tablet is not the same as the printed page nor is it the same as the blackboard, the TV screen, or that of other media; it is unique.

We can aid reading in many ways. We can make extensive use of blank space, since it is free on the computer screen. We can use short lines. We can avoid hyphenation. We can divide text so that natural phrases stay together on a line. We can remove information no longer relevant. We can interweave text and graphics.

In addition to these factors, we can use timing in a variety of ways, often for the same purposes. We can pause before or after an important word or phrase. We can pause at natural grammatical breaks.

All these tactics are used not only in the present material but are also used in other material developed at the Educational Technology Center. Further information is available on request. While our use of these factors is based largely on informal research, over a fourteen year period, we now are conducting more formal studies of some of these problems.

Descriptions of Individual Units

This section describes the units developed in this project. Detailed papers are available.

A. The Measurement Process

Two of the units concern the nature of the measurement process in science. Since measurement is at the heart of science, it is important to understand this empirical basis of the scientific activity. We need to explicate the nature of operational definitions and how terms in theories are connected with experience through measurement operations.

1. Distance. One unit is based on the concept of distance. This subject was chosen because it is a measurement that almost all students already know how to make. We raise many major issues about the measurement process, illustrating them in a situation in which almost everyone is already partially familiar with the process involved.

The module starts with simple measurements of distance. Each student station is equipped with a ruler and a board with several dots marked on the board and labeled. The student measures some distances on the board.

The student is made conscious of the processes involved in measurement, the ingredients of an operational definition. The idea that not everybody makes the same measurement in a given situation receives considerable attention in the remainder of the unit. The student started out to measure "the distance," but each person seems to have his or her "own" distance. This common feature of science leads to discussion of average and standard deviation and sheds light on the nature of the measurement process.

Another issue raised is that of different ways of measuring "something" which still is called distance. The issue is raised as to why each of these very different proceses are said to measure the "same thing."

2. Area. A computer dialog about area continues many of these arguments. The nature of area, particularly the area of a general figure, is likely to be unknown to the users. So an important aspect of this dialog is to build with full input from the student an adequate concept of area to cover all situations involving closed planar curves.

This unit begins with an attempt to improve intuitive notions of area. A student is given an irregularly shaped area and a square which can grow or shrink and is asked to make the area of the square the "same" as the area of the other figure. This activity is carried out before a formal definition of area, so it is an intuition using and developing operation. The role of intuition is an important one in many of the other activities.

B. Discovery of Scientific Theories

In this group of dialogs, students are in the role of a scientist. They gather evidence, develop hypotheses, test these against the evidence, and modify them. The interactive environments are protective of the students--they will eventually succeed if they continue to work.

1. Optics. This unit is intended for this project and for another project. It concerns the law of reflection for light. As with many of the units, it is a discovery module. The student has an empirical capability, a flashlight that could light a target through a mirror. The early hypotheses developed do not work when new evidence is developed and must be modified through later experimentation. One section isolates which factors might be involved in the question of where to shine the light.

2. Tribbles. Again, an empirical facility, the ability to observe the changing patterns of the Tribbles on the planet Trib is at the heart of this dialog. Students observe Tribble patterns, and then are encouraged to create laws governing the development of the Tribble patterns. The patterns are based on the game of Life as developed by John Conway and made well known by Martin Gardner's discussions in Scientific American. The emphasis is on prediction, with each rule developed; using the rule the student predicts the next generation pattern. Then this prediction is compared with the actual pattern. The environment is conducive to eventually discovering the correct laws, because of help and aid given to people in trouble. Thus, the experience will turn out to be successful for almost all users.

3. Tribble Families. Although we again use the name Tribbles, this program really does not have much to do with the previous one. Here we observe the offspring of entities called Tribbles, again on a remote planet. Tribbles differ only in three characteristics. Again everything must be determined by working empirically through the evidence, just as in the previous dialog. The basic model followed is the genetic model; Tribble characteristics are inherited just like those of people and animals on earth.

4. Electricity. This dialog is based on the ESS unit on batteries and bulbs. The initial experiential problem is simply lighting a light bulb, given a battery, some wires, and the bulb. The student is given great freedom with the program only gradually beginning to give very selected assistance based on just what the student has been doing. Again, the aim is to develop a scientific model through empirical studies in a slow and careful fashion, checking understanding carefully at each stage. Although the content is that of simple circuit theory, the learning objectives are quite different with emphasis on the way the scientist would go about studying the situation and developing the necessary model.

Current Status

All these dialogs have undergone extensive field testing in recent months. Although our major location was the public library, we have also tested the material in science museums, junior high schools, high schools, community colleges, and universities. In one case (University of West Virginia) some of the materials are being used to structure a complete undergraduate course in science.

Based on this field testing and the information gathered, we are now revising the material. Many of these revisions are complete, while others are still underway as of this writing.

COMPUTING FOR THE HANDICAPPED

Paul L. Hazan
Johns Hopkins University
Applied Physics Laboratory
Laurel Md, 20707

The evolution of low-cost personal computing technology promises unprecedented educational and vocational opportunities for millions of handicapped citizens. Extending the reach and improving the quality of life of disabled people are surely among the most worthwhile achievements that will result from the expanding use of computers. A promising new alliance between the power of computing and the urgent needs of the handicapped is providing a new impetus that is capturing the imagination of both the handicapped and the computer communities.

In sympathy with the need to relieve the constraints under which many handicapped people live and to turn opportunities into reality, the Johns Hopkins University First National Search was conceived as a competition to inspire computer-based applications aimed at meeting the educational, vocational, daily living, and recreational needs of disabled persons.

Objectives of the National Search

The objectives of the National Search were to:

- Focus the power of computing technology on the urgent needs of millions of handicapped individuals
- Stimulate individual innovation and creativity in the general public, and
- Encourage individuals, professional societies, and academic, industrial, government, civic and rehabilitation organizations to work together to meet a major human need.

The Need

The number of people afflicted with mental and physical handicaps typically represents approximately 10% of a country's population. In the United States, for instance, this figure is in excess of 25 million. This includes individuals with handicaps related to vision, hearing and language, mental retardation, learning disabilities, neuromuscular or neurological disorders, and limitations of movement.

There are, for example, millions of children suffering from learning disabilities whose education may be significantly helped through the innovative application of available personal computing technology. The addition of such standard modules as voice synthesizers and voice recognition devices promises to extend the reach and scope of activities of many persons suffering from visual, communication, and mobility handicaps. Many entries submitted to the Johns Hopkins National Search provide innovative and practical solutions to educational and vocational challenges promising to improve the quality of life of countless disabled individuals. Such solutions may convert dependence on others to independence and increase the productivity of handicapped people on an unprecedented scale. Individual innovation coupled with the leverage of personal computing can transform the dedication and drive of millions of handicapped individuals into a major national resource.

The Competition

The Johns Hopkins National Search sought entries from three classes of participants:

a) Computer Professionals

b) Amateurs (This included professionals in non-computer related fields)

c) Students

Team entries were encouraged. Many team entries involved three participants, e.g. a computer professional, a rehabilitation specialist, and a handicapped person.

Submissions were invited in the following categories:

o Working computer based devices

o Working computer programs

o Paper designs (i.e., concepts not yet implemented)

The use of standard and general purpose building blocks was encouraged to facilitate subsequent availability to handicapped users. It was emphasized that the general purpose nature of computers coupled with the flexibility of individualized software programs could offer the handicapped user a range of personalized tools while retaining the advantages of mass production and distribution.

Fifteen introductory workshops were held around the country to bring together technologists, who had the potential tools and the handicapped community, who had and understood the needs. The workshops were held at locations such as The Maryland Rehabilitation Center, Gallaudet College for the Deaf, NASA's Johnson Space Center and the Trace Center in Madison, Wisconsin. Participants ranged from computer scientists and hobbyists to rehabilitation professionals, educators, and the handicapped themselves. The public response to the national search and to the workshops was enthusiastic. Thousands of individuals and many institutions interested in problems of the handicapped responded supportively with a wide range of innovative ideas and applications.

Over 900 competition entries were received and several hundred of these were exhibited at 10 regional fairs around the country. The fairs were held in August 1981 at major centers such as the Chicago Museum of Science and Industry, the San Francisco Exploratorium, and the Boston Museum of Science.

Competition entries were evaluated and judged on the basis of technical merit and functional promise. The evaluation was done by experienced judges from technical, academic and rehabilitation organizations, as well as by many handicapped individuals. Over 100 regional awards, including 26 personal computers and many cash prizes, were awarded to winning regional entries. From these, the national finalists were selected and invited to exhibit their inventions in the Great Hall of the National Academy of Sciences in Washington DC, and to compete for the national awards, including a $10,000 grand prize.

Twenty-eight finalists exhibited their inventions at the National Fair. Handicapped people, educators, rehabilitation specialists, technologists, and the general public attended the Fair in large numbers, and extensive media coverage took the event to radio and television audiences.

The national judging was held on November 1st, 1981, one year after the launching of the program at the National Press Club. The nineteen national judges were faced with the challenging task of making a final selection from an impressive array of innovative and practical entries. Professionals, amateurs, and students were represented among the national finalists. The Awards Banquet, held the next evening, was attended by 300 guests who were addressed by leaders of the handicapped, educational, technical, and government communities.

This was followed by a two-day workshop bringing together the inventors, potential users, industry, government agencies, educators and professional societies interested in problems of the handicapped.

The workshop was held at The Johns Hopkins University Applied Physics Laboratory. It was highly interactive and included working exhibits as well as presentations by the national finalists. Paper topics ranged from a reading machine for the blind to portable telecommunications for the deaf using a pocket computer which couples to a standard telephone. Some applications involved computer aided instruction for the learning disabled, while others provided a quadriplegic with an ultrasonic head control for a wheelchair or an electronic keyboard with word-processing capability. To recognize and further publicize the achievements of the regional winners, Proceedings describing 97 regional winning submissions were publishd in cooperation with the IEEE Computer Society.

Conclusions

The personal computing community has been accused by some of producing little more than expensive games. Others have predicted far reaching opportunities and implications for a broad spectrum of society from this ubiquitous technology. Much has been said about the potential of personal computing for helping the handicapped, and some successes were achieved in the past. For many, however, the Johns Hopkins First National Search provided an exceptional opportunity to translate creative ideas into reality. Over 1000 innovative individuals rose to the challenge.

The task of improving the quality of life, as well as the educational and vocational opportunities for the millions of handicapped people, is awesome indeed. But then so is the power of individual creativity. We believe that the store of individual ingenuity and dedication coupled with the creative leadership of our educational, technological and rehabilitation communities makes this ambitious goal not only attainable but inevitable.

Appendix: List of Regional winning entries by handicap.

Reference: "Computing and the Handicapped: The Johns Hopkins First National Search." INFOTECH State of the Art Conference, London November 1981

PAUL L. HAZAN is Assistant to the Director for Advanced Computer Technology at The Johns Hopkins University Applied Physics Laboratory. He is program director of the Johns Hopkins First National Search for the Application of Personal Computing to Aid the Handicapped.

Hazan served on the Governing Board of the IEEE Computer Society from 1978-81 and as chairman of the Society's Mini/Micro Technical Committee. He has lectured on microcomputer technology at the graduate level and is the author of mumerous papers on computing and has chaired several national and international computer conferences, including COMPCON Fall 77.

Hazan obtained his BSc in electrical engineering in 1952 from the Royal College of Science and Technology in Britain and did his graduate work in computer science at the University of Maryland.

TITLE INDEX

A COMPUTER LITERACY PROGRAM FOR ELEMENTARY AND MIDDLE SCHOOL CHILDREN AND TEACHERS
Halapin, Jack 38

A MICROCOMPUTER-BASED INSTRUCTIONAL MANAGEMENT SYSTEM
Rodel, Lee J. 206

A PROBLEM SOLVING AUTHORING SYSTEM
Perry, Ted 47

A SUPERINTENDENT'S PERSPECTIVE ON MANAGEMENT INFORMATION SYSTEMS
McFatter, William 360

ADDRESSING THE NEED FOR PRE-COLLEGE TEACHER TRAINING IN COMPUTER SCIENCE IN THE STATE OF GEORGIA
Clark, Dr. Frank C. and Carter, Dr. Harrison S. 412

ALGORITHMS AND THEIR SOURCE CODE IMPLEMENTATION
Nuesse, David A. 198

AN ON-LINE LIBRARY CIRCULATION SYSTEM
Pennington, Billy 457

AND THE TWAIN SHALL MEET: COMPUTERS AND THE TEACHING OF ENGLISH
Shostak, Robert 55

APPLIED DISCRIMINANT MODEL FOR INFORMATION SYSTEMS EVALUATION
Rushinek, Avi; Rushinek, Sara F.; Stutz, Joel D. 389

APPLYING COMPUTERS EFFECTIVELY IN THE SMALL TWO-YEAR COLLEGE
Conrad, Lora P. 192

APPLYING INSTRUCTIONAL AND COMPUTER TECHNOLOGY TO THE DEVELOPMENT OF IEP'S: PROJECT RECIPE
Reid, George A., Jr. and Bell, H. Sanders, III 330

BASICS, NOT BASIC: OR TEACHING WRITING TO THE COMPUTER
Stuart, Dr. Ann 337

BIG PROBLEM - STUDENT SCHEDULING AND GRADE REPORTING, BIG SOLUTION - THE MICROCOMPUTER
Mosow, Dr. David K.; Bolton, Brenda; Young, Henderson 385

BUILDING AN EFFICIENT AND EFFECTIVE MANAGEMENT INFORMATION SYSTEM
Marshall, Jon C. 428

CAI: PRACTICE, ART, AND SOMETIMES SCIENCE
Burke, Robert L., Ed.D. 34

CALCULUS TUTORIAL SYSTEM
Arganski, Nicholas 123

CHANGE-LIKE IT OR NOT!
Wallisch, Lt. Col. (Dr.) Bill 217

COGNITIVE STYLE AND A FIRST COURSE IN COMPUTER SCIENCE: A SUCCESS STORY
Hassell, Johnette 163

COMPREHENSIVE DATA MANAGEMENT SYSTEM (CDMS) FOR INDIVIDUAL EDUCATION PLANS AND SPECIAL EDUCATION PROGRAM MANAGEMENT
Guth, Robert O., Ed.D. 304

COMPUTER EDUCATION - A PRELIMINARY PLAN OF ACTION FOR THE DADE COUNTY PUBLIC SCHOOLS
Olkes, Alan T. 436

COMPUTER LITERACY EDUCATION FOR K-6 TEACHERS
Philipp, Charles C.; Muntner, Judie; Cutlip, Patricia 322

COMPUTER LITERACY FOR EDUCATIONAL ADMINISTRATORS
Spero, Samuel and Jonas, Jan 292

COMPUTER LITERACY IN THE ELEMENTARY SCHOOL: AN ARGUMENT FOR CHANGE FROM WITHIN
Winner, Alice Ann 265

COMPUTER LITERACY TRAINING NEEDS OF INSERVICE AND PRESERVICE TEACHERS: IMPLICATIONS FOR TEACHER EDUCATION PROGRAMS
Anderson, Cheryl 119

COMPUTER LITERACY - TEACHER EXPECTATIONS AND STUDENT ACHIEVEMENTS
McCabe, Jack 290

COMPUTER POWER: A COORDINATED SERIES OF HIGH SCHOOL CURRICULA
Hughes, Charles E. and Moshell, J. Michael 350

COMPUTER SIMULATION LABORATORY FOR DESIGN OF EXPERIMENTS
Hirsch, R. A.; Geremia, J. O.; Wu, C. 421

COMPUTER SIMULATIONS STIMULATE INDUCTIVE THINKING
Gress, Eileen K. 445

COMPUTER SUPPORTED STUDENT MANAGEMENT FOR A LARGE CLASS
Boyle, Thomas A. 156

COMPUTERS+BOOKS+KIDS=REAL LEARNING
Wall, Elizabeth S. 465

COMPUTER-ASSISTED TEACHING OF ARABIC
Johnson, J. Matthew and Bonney, William W., Jr. 283

COMPUTER-MORPHOSIS
Weaver, Sister Catherine, O.S.F. 469

COMPUTING FOR THE HANDICAPPED
Hazan, Paul L. 476

COST EFFECTIVE COMPUTER GRAPHICS TERMINAL UTILIZATION: THE STUDENT AS TEACHER
Brillhart, L. V. and Bell, Eric A. 236

COURSEWARE CRITERIA FROM AN INSTRUCTIONAL DESIGN PERSPECTIVE
Roblyer, M. D. 51

CREATIVE PROGRAMMING FOR YOUNG MINDS
Buxton, Marilyn and Taitt, Henry A. 241

CRITERIA FOR ESTABLISHING A COMPREHENSIVE COMPUTER PROGRAM FOR A SMALL LIBERAL ARTS COLLEGE
Wilde, Jane H. 71

DECENTRALIZING SCHOOL ADMINISTRATIVE SUPPORT SERVICES WITH SMALL BUSINESS MICROCOMPUTERS IN A LARGE SCHOOL SYSTEM
Bybee, Dennis L., Ph.D. 277

DESIGNING INSERVICE COMPUTER LITERACY TRAINING FOR TEACHERS
Martin, C. Dianne and Heller, Rachelle S. 15

DEVELOPING A FACULTY COMPUTER LITERACY PROJECT IN A MULTI-CAMPUS COMMUNITY COLLEGE
Guthrie-Morse, Barbara, Ph.D. 160

DEVELOPMENT OF A STATEWIDE SYSTEMS ARCHITECTURE: EXPERIENCES FOR THE STATE OF MINNESOTA
Bahn, Theodore 147

DISTRIBUTED DATA PROCESSING: GENERAL CONSIDERATIONS AND A USER EXPERIENCE
Umar, Amjad 59

EDUCATIONAL COMPUTING: SUPPORT FUNCTIONS AND STUDENT OBJECTIVES
Bybee, Dennis L., Ph.D. 80

EFFECTIVENESS OF COMPUTER-BASED INSTRUCTION IN JOB CORPS
Geller, Daniel M., Ph.D. 134

EFFECTS OF PROGRAMMING ON MATHEMATICS ACHIEVEMENT
Harrison, Mary O. 416

ESTABLISHING A HARD-DISK BASED MICROCOMPUTER NETWORK
Tinsley, Jack T. 339

EVALUATING AND SELECTING A BUSINESS BASIC
Hoey, James M. 311

EVALUATION OF THE IMPLEMENTATION AND MANAGEMENT OF MICROCOMPUTERS FOR INSTRUCTION
Van Dusseldorp, Ralph 296

EVALUATION RESEARCH - A CATALYST FOR CHANGE IN COMPUTING
Lee, L. D. and Haughey, D. G. 178

FACTORS RELATING TO THE UTILIZATION OF INFORMATION BY SUPERINTENDENTS FOR DECISION-MAKING
Somers, Brenda 326

FLORIDA CENTER FOR INSTRUCTIONAL COMPUTING
Engel, Dr. C. William 83

FLORIDA CHOICES - HELPING STUDENTS CHOOSE
Thomas, Glenn and Wooley, William 396

FLORIDA'S INFORMATION RESOURCE NETWORK
Olkes, Alan T. 98

FULFILL INSTRUCTIONAL COMPUTING NEEDS WITH VARIOUS MICROCOMPUTERS
Brumbaugh, Dr. Kenneth E. 300

HIGHER EDUCATION'S RESPONSIBILITY TO JOB TURNOVER IN COMPUTER-RELATED JOBS
Bailey, M. Gene and Black, David 127

IDENTIFICATION AND EVALUATION OF EDUCATIONAL APPLICATIONS SOFTWARE
Bockmann, Fred 9

IMPLEMENTING INSTRUCTIONAL COMPUTING AND COMPUTER LITERACY IN A SCHOOL OR COLLEGE
Bell, Frederick H. 406

IMPROVING THE VALIDITY OF COMPUTERIZED MULTIPLE-CHOICE TESTS IN MATHEMATICS
McHenry, Dr. Hugh 19

INSTRUCTIONAL MANAGEMENT: CIMS-IMPACT CASE STUDY
Adams, Charles and Cameron, Kathryn 143

INSTRUCTIONAL USES OF COMPUTERS: THE GOOD, THE BAD, AND THE UGLY
Sloan, Sally A. 23

INTEGRATING CAI AND VIDEOTAPE FOR SCIENCE AND MATHEMATICS
Ignatz, Mila E. 196

INTEGRATING MICROCOMPUTERS INTO THE MATHEMATICS CURRICULUM--USING THE COMPUTER AS AN ADJUNCT INSTRUCTOR
Schwandt, Lynn C. 258

INTERACTIVE LANGUAGE EXPERIENCES BETWEEN THE COMPUTER AND THE STUDENT
Kuchinskas, Gloria 12

IT DOES NOT HAVE TO BE A BLACK BOX: THE CREATIVE USE OF A MICROCOMPUTER AS A COMPUTATIONAL AID IN THE EDUCATIONAL ENVIRONMENT
Pas, Eric I. and Anapol, Edward 102

LEFT-HANDED LEARNERS, THEIR HEMISPHERICITY PROBLEMS AND COMPUTERS
Donoian, George, Ed.D. and Donoian, Eleanor, BA 381

Donoian, George, Ed.D. and Donoian, Eleanor, BA 381

LEVELING COURSES FOR STUDENTS PURSUING A MASTERS DEGREE IN COMPUTER SCIENCE
Swigger, Dr. Katheleen M. and Zamecnik, Dr. Fred J. 106

LONG RANGE PLANNING FOR COMPUTER USE - A TASK FORCE MODEL
Raucher, S. M. and Koehler, T. J. 367

MANAGING A MAJOR RESOURCE - MANAGEMENT AND INSTRUCTIONAL INFORMATION
Somers, Brenda 261

MANAGING INFORMATION FOR NEW KINDS OF LEARNERS
Markovich, Greg and Malling, Joan 319

MATCHING FUNCTIONS OF THE APPLE COMPUTER TO THE INSTRUCTIONAL NEEDS OF THE LEARNING DISABLED STUDENT
Archer, Doug 1

MATCHING INSTRUCTIONAL PROGRAMS TO LEARNING PROBLEMS
Pollard, James, Ed.D. 203

MICROCOMPUTER APPLICATIONS IN THE COMMUNITY COLLEGE
Dellow, Dr. Donald A. and Poole, Dr. Lawrence H. 225

MICROCOMPUTER APPLICATIONS IN THE SPECIAL EDUCATION CLASSROOM
Turner, Greg 400

MICROCOMPUTER INSTRUCTIONAL SUPPORT IN THE CLASSROOM (MISIC) PROJECT - MICROCOMPUTERS IN THE CLASSROOM K-12 A "HOW TO DO IT" REPORT
Becht, Paul and Becht, Sara-Maria F. 32

MICROCOMPUTER PATIENT EDUCATION COURSEWARE
Canine, Kent, Ph.D. and McWeeny, Mae 244

MICROCOMPUTER SOFTWARE: EVALUATE--EVALUATE--EVALUATE
Taber, Florence 138

MICROCOMPUTERIZED KAUFMAN WISC-R HYPOTHESIS GENERATION
Ingram, Rich 92

MICROCOMPUTERS IN COOPERATIVE GROUPS
Corens, Ken and Maggie 247

MICROCOMPUTERS IN SPECIAL EDUCATION--BOON OR BANE?
Hannaford, Alonzo E. 250

MICROCOMPUTERS IN THE GIFTED/TALENTED PROGRAM
Berry, Mary 131

MICROCOMPUTERS: ADMINISTRATIVE TIMESAVERS
Kehrer, Gary A. and Schepia, Nick A. 94

MICROCOMPUTERS: GETTING STARTED IN A JUNIOR HIGH SCHOOL
Pinski, Mary M. 364

OBSERVATIONS ON THE DYNAMICS OF K-12 TEACHER INVOLVEMENT WITH MICROCOMPUTERS
Judd, Dorothy H., Ed.D. and Judd, Robert C. 315

OCCUPATIONAL PROFICIENCY PREPAREDNESS OR FUTURE SHOCK?
Adams, Sue 377

OUR ADVENTURES WITH THE COMPUTER
McClellan, Jane and Griffin, Holley 432

PROCEDURES FOR DEVELOPING ADMINISTRATIVE SOFTWARE
Loewen, Duane 229

PROJECT CAISH: CAI AND SUPPORT FOR THE HANDICAPPED
Brown, Warren 273

REINFORCING READING SKILLS USING THE MICROCOMPUTER AND SPECIALIZED SOFTWARE DESIGN BY ELEMENTARY AND MIDDLE SCHOOL TEACHERS
Lagotic, Diana, Ph.D; Losch, Lorraine; Domenech, Lynn 176

SCIENCE LITERACY IN THE PUBLIC LIBRARY
Bork, Alfred; Franklin, Stephen; Von Blum, Ruth; Trowbridge, David;
Kuntz, Barry L. 473

SELECTING EDUCATIONAL APPLICATIONS FOR MICROCOMPUTERS
Hausmann, Linda Borry 88

SOCIAL ISSUES IN COMPUTING: A COURSE FOR COMPUTER SCIENCE MAJORS
Bailey, T. E. 5

SOFTWARE SHARING IN FLORIDA COMMUNITY COLLEGE BURROUGHS USERS CONSORTIUM
Monish, A. R. 453

STOREROOMS OR CLASSROOMS? THE FUTURE OF MICROCOMPUTERS IS IN THE TRAINING OF ELEMENTARY SCHOOL TEACHERS
Richard, Dr. Tom 184

STUDENT SELF-TRACKING USING A MICROCOMPUTER
Strong, Mary Winifred 334

STUDY TO EXAMINE COMPUTER TECHNOLOGY FOR IMPLEMENTATION INTO A COLLEGE READING PROGRAM
Williams, Carmelita 140

SUCCESSFUL IMPLEMENTATION OF COMPUTING TECHNOLOGY IN SCHOOLS THROUGH TEACHER EDUCATION, PROFESSIONAL DEVELOPMENT AND SUPPORT
Willis, Cam 75

TAXONOMY OF INSTRUCTIONAL STRATEGIES FOR COMPUTER BASED EDUCATION
Hall, Keith A.; Comer, Ronald C.; Merrill, John A. 307

TEACHING YOUNG CHILDREN TO PROGRAM IN BASIC
VanHorn, Royal 28

TECHNIQUES FOR USING MICROCOMPUTERS TO AID LEARNING DISABLED AND
MENTALLY RETARDED STUDENTS
Mason, Marguerite 286

THE ABC'S OF FARMING OR AGRICULTURE BY COMPUTER
Hintz, Thomas R. 167

THE APPLICATION OF DISTRIBUTED PROCESSING AND NETWORKING AT BROWARD
COUNTY SCHOOLS TO MANAGE STUDENT RECORDS AND PERFORM INSTRUCTIONAL
MANAGEMENT
Kauffman, Fred 171

THE COMPUTER IN THE SCHOOLS -- WHAT'S BEEN SAID TO THE TEACHER?
Saltz, Martin 393

THE COMPUTER: AN AFFECTIVE MACHINE
Roser, Jeff 461

THE DANGER OF THE REDUNDANT MICRO
Vinlove, Donald J. 403

THE DESIGN OF A STRUCTURED CAI LANGUAGE AND CAI SUPPORT SYSTEM
Harris, Marshall 346

THE DESIGN, DEVELOPMENT AND EVALUATION OF A LOW-COST COMPUTER MANAGED
SPELLING SYSTEM
Thomas, Rex and Gustafson, Brian 110

THE IMPACT OF MICROCOMPUTING ON SECONDARY SCHOOL ADMINISTRATORS
Bowers, Larry 152

THE NUTS AND BOLTS OF SELECTING AND MAXIMIZING COMPUTER ASSISTED
INSTRUCTION
Conkling, Richard D. 188

THE POLITICAL PLATFORM OF THE COMPUTER
Leslie, Judith W. 356

THE USE OF GEOPROCESSING IN EDUCATIONAL RESEARCH AND DECISION SUPPORT
Sexton, Porter 232

TO COMPUTERIZE OR NOT? A QUESTION OF GIGO!!
Zuckerman, Robert A., Ed.D. 342

TOWARD A FUNCTIONAL INTEGRATION OF COMPUTERS IN THE EDUCATIONAL
PROCESS: A MICROCOMPUTER APPLICATION FOR FACILITATING MEMORIZATION
Schilmoeller, Kathryn J; King, Dennis R.; Schilmoeller, Gary L. 213

TRANSPARENT SORTS BETTER SERVE USERS
Muldrow, William R. 254

UNDERSTANDING THE "CAI PHENOMENON"
Bright, George W. ... 221

USING A MICROCOMPUTER TO MANAGE LARGE FILES OF INSTRUCTIONAL PICTURES
Price, Samuel T. and Dillingham, Larry M. ... 440

USING COMPUTER GRAPHICS AS A CRITICAL COMPONENT IN MATHEMATICS INSTRUCTION
Flake, Janice L. ... 279

USING MICROCOMPUTERS TO IMPROVE YOUR HEARING IMPAIRED PROGRAM
Sarapata, Susan L. ... 374

USING THE MICROCOMPUTER TO APPLY DATA-BASED DECISION RULES FOR ASSISTING IN THE SELECTION OF INSTRUCTIONAL STRATEGIES FOR THE HANDICAPPED
Hasselbring, Ted S. and Hamlett, Carol L. ... 43

VOCATIONAL TRAINING FOR HIGH SCHOOLS IN THE AGE OF THE MICRO
Lorton, Paul, Jr. and Muscat, Eugene J. ... 450

WHAT IS REAL COURSEWARE?
Rohm, Robert R. ... 210

WHAT SHOULD BE INCLUDED IN COMPUTER LITERACY TRAINING FOR TEACHERS?
Andrews, Bernice Doerr ... 269

WORD PROCESSING/TEXT EDITING USING MICROCOMPUTERS
Wetterlind, Dr. Peter ... 115

AUTHOR INDEX

Adams, Dr. Charles F.
Oneida-Madison-Herkimer BOCES
Box 70
New Hartford, NY 13413

Adams, Sue
Tarrant Co. Jr. College
NE Campus/828 Harwood Rd.
Hurst, TX 76053

Anapol, Edward
Duke Univ.
Dept. of Civil Eng.
Durham, NC 27706

Anderson, Dr. Cheryl A.
Univ. of Texas at Austin
406 Education Bldg.
Austin, TX 78712

Andrews, Dr. Bernice
Univ. of Central Fla.
P.O. Box 25000/Dept. of Comp. Sci.
Orlando, FL 32816

Archer, Doug
Iowa Reg. Comp. Consortium
Green Valley Rd./Green Val. AEA 14
Creston, Iowa 50801

Arganski, Nicholas
P.O. Box/CP 2000
Ste Anne De Bellevue, Quebec H9X 3L9

Askalani, Mohamed H. Ph.D.
Mankato St. Univ.
Ctr. for Economic Ed., Box 14
Mankato, MN 56001

Bahn, Theodore I.
Peat, Marwick, Mitchell & Co.
1700 IDS Ctr.
Minneapolis, MN 55402

Bailey, Gene
Western Carolina Univ.
Cullowhee, NC 28723

Bailey, T. E.
Dept. of Admin. Sciences
Univ. of Central Arkansas
Conway, Arkansas 72032

Becht, Dr. Paul A. and Sara-Marie F.
Univ. of Florida
P.K. Yonge Lab. School
Gainesville, FL 32611

Bell, Eric A.
Triton College
2000 Fifth Ave.
River Grove, IL 60171

Bell, Frederick H.
Univ. of Pittsburgh
4801 Forbes Quad
Pittsburgh, PA 15260

Bell, Sanders
Sarasota Co. Sch. Board
4747 S. Tamiami Trail
Sarasota, FL 33581

Bender, Louis W.
Florida State Uuniversity
113 Stone Bldg.
Tallahassee, FL 32306

Berry, Dr. Mary N.
815 Crest Dr.
Omaha, NE 68046

Black, David
Western Carolina Univ.
Cullowhee, NC 28723

Bockman, Fred
Project Micro-Ideas
2941 Linneman
Glenview, IL 60025

Bolton, Brenda
Davidson High School
Mobile, Alabama

Bonney, William W., Jr.
Interactive Research Assoc.
P.O. Box 950
Iowa City, Iowa 52240

Bork, Alfred
Univ. of California
Educational Tech. Ctr.
Irvine, CA 92715

Bowers, Larry
N. High School
4436 Main St.
Downers Grove, IL 60515

Boyle, Thomas
Div. of Interdis. Eng. Studies
Purdue Univ.
W. Lafayette, Indiana 47907

Bright, George W.
Northern Illinois Univ.
Dept. of Math Sci.
Dekalb, IL 60115

Brillhart, Dr. L. V.
Triton College
2000 Fifth Ave.
River Grove, IL 60171

Brown, Warren R.
Sarasota Co. Sch. Board
3450 Gocio Rd.
Sarasota, FL 33580

Brumbaugh, Dr. Kenneth E.
Minnesota Ed. Comp. Consortium
2520 Broadway Dr.
St. Paul, MN 55113

Burke, Robert L.
Nova Univ.
3301 College Ave.
Ft. Lauderdale, FL 33314

Buxton, Marilyn
Creative Programming
604 6th St.
Charleston, IL 61920

Bybee, Dr. Dennis L.
Dept. of Defense Depend. Schs.
2461 Eisenhower Ave./Hoffman Bldg. 1, 148
Alexandria, VA 22331

Cameron, Kathryn
Box 70
New Hartford, NY 13413

Canine, Kent
Sister Kenny Institute
2727 Chicago Ave. S.
Minneapolis, MN 55407

Carter, Harrison S.
Georgia Southern College
Landrum Box 8093
Statesboro, GA 30458

Clark, Frank C.
Georgia Southern College
Landrum Box 8093
Statesboro, GA 30458

Comer, Dr. Ronald C.
Ohio State Univ.
Columbus, OH 43210

Conkling, Richard D.
Bunnell Elem. Sch.
P.O. Box 462
Bunnell, FL 32010

Conrad, Lora
Florida State University
113 Stone Bldg.
Tallahassee, FL 32306

Corens, Ken and Maggie
Evaluation Ctr.
1001 State Highway 7
Hopkins, MN 55343

Cutlip, Patricia
850 Hungerford Dr.
Rockville, MD 20850

Dellow, Dr. Donald R.
N. Country Comm. Col.
Saranac Lake, NY 12983

Domenech, Lynn
Alachua Co. Schools
1702 NW 38 Dr.
Gainesville, FL 32605

Donoian, Dr. George & Dr. Eleanor
Detroit Public Schools
10001 Puritan
Detroit, MI 48238

Engel, Dr. Charles W.
Univ. of S. Fla.
Col. of Ed., EDU 307H
Tampa, FL 33620

Flake, Janice L.
1065 Merritt Dr.
Tallahassee, FL 32301

Geller, Daniel
1625 Eye St., NW
Suite 510
Washington, DC 20006

Geremia, J.O.
U.S. Naval Academy
Mechanical Eng. Dept.
Annapolis, MD 21402

Gress, Eileen K.
Fairport Central Sch.
140 Hulburt Rd.
Fairport, NY 14450

Griffin, Holley
Ft. King Middle Sch.
545 NE 17th Ave.
Ocala, FL 32670

Gustafson, Brian
Iowa St. Univ.
Computation Ctr.
Ames, Iowa 50011

Guth, Dr. Robert O.
Fairfield Public Schools
P.O. Box 220
Fairfield, Conn. 06430

Guthrie-Morse, Dr. Barbara
P.O. Box 3028
Valencia Comm. Col.
Orlando, FL 32804

Halapin, Jack
Ridgefield Public Schools.
Box 658
Ridgefield, CT 06877

Hall, Dr. Keith A.
Ohio St. Univ./121 Ramseyer Hall
29 W. Woodruff Ave.
Columbus, OH 43210

Hamlett, Carol L.
North Carolina St. Univ.
402 Poe Hall
Raleigh, NC 27650

Hannaford, Alonzo
Western Michigan Univ.
3616 Edinburgh
Kalamazoo, MI 49007

Harris, Marshall
Dept. of Computer Science
Univ. of Queensland
St. Lucia, Queensland, Australia 4067

Harrison, Mary O.
Univ. of Central Fla.
1015 Sapling Dr.
Winter Spgs., Fl 32708

Hasselbring, Ted S.
NC St. Univ.
402 Poe Hall, Box 5096
Raleigh, NC 27650

Hassell, Ph.D., Johnette
Tulane Univ.
Sch. of Eng./301 Stanley Hall
New Orleans, LA 70118

Haughey, Dan
Manitoba Dept. of Ed.
408-1181 Portage Ave.
Winnipeg, Manitoba, Canada R3G0T3

Hausmann, Linda Borry
1750 26th Ave. NW
St. Paul, MN 55112

Hazan, Paul L.
Johns Hopkins Univ.
Applied Physics Lab.
Laurel, MD 20707

Heller, Rachelle S.
Univ. of Maryland
Dept. of Com. Sci.
College Park, MD 20742

Hintz, Thomas R.
Univ. of Fla.
1022 McCarty Hall
Gainesville, FL 32611

Hirsch, R.A.
U.S. Naval Academy
Mechanical Eng. Dept.
Annapolis, MD 21402

Hoey, M. James
Potomac St. Col. of WV
Box 567
Keyser, WV 26726

Hughes, Charles E. & Moshell, J. Micheal
Univ. of C. Fla.
Comp. Sci. Dept.
Orlando, FL 32816

Milagros E. Ignatz
Fla. A&M Univ.
Box A-19/Lab Sch.
Tallahassee, FL 32307

Ingram, Rich
Jefferson Co. Schools
110 Mordington Ave.
Charles Town, WV 25414

Johnson, J. Matthew
Interactive Res. Assoc.
P.O. Box 950
Iowa City, Iowa 52240

Judd, Drs. Robert C. & Dorothy H.
1990 Flagstaff Ct.
Glendale Heights, IL 60137

Kauffman, Fred H.
Broward Co. Sch. Bo.
1300 W. Las Olas Blvd.
Ft. Lauderdale, FL 33312

Kehrer, Gary
Broward Co. Sch. Bo.
1320 SW 4th St.
Ft. Lauderdale, FL 33312

King, Dennis R.
Univ. of Maine
Human Devel.
Orono, Maine 04469

Koehler, Mr. Tommas
Montgomery Public Schs.
850 Hungerford Dr.
Rockville, Md 20853

Kuchinskas, Dr. Gloria
P.O. Box 4172
Tallahassee, FL 32303

Lagotic, Dr. Diana
Alchua Co. Schools
1702 NW 38 Dr.
Gainesville, FL 32605

Lee, Linda
Manitoba Dept. of Ed.
408-1181 Portage Ave.
Winnipeg, Manitoba, Canada R3GOT3

Leslie, Judith W.
Pima Comm. Col.
2202 W. Anklam Rd.
Tucson, AZ 85709

Loewen, Duane
MECC
Computer Dept. MSU
Moorhead, MN 56560

Lorton, Dr. Paul, Jr.
Univ. of San Francisco
San Francisco, CA 94117

Losch, Lorraine
Alachua Co. Schools
1702 NW 38 Dr.
Gainesville, FL 32605

Malling, Joan
Marylhurst College
Erie, PA 16546

Markovich, Greg
Mercyhurst College
Glenwood Hills
Erie, PA 16546

Marshall, Jon C.
Univ. of Missouri
AFSE Dept.
St. Loouis, MO 63121

Martin, Dianne C.
Univ. of Maryland
Dept. of Computer Science
College Park, MD 20742

Mason, Marguerite & Walter Bruce
Proj. Mass/Comm. Ut. Sch. Dist. 1
410 W. Polk
Charleston, IL 61920

McCabe, Jack
Mainland Sr. High
125 S. Clyde Morris Blvd.
Daytona Bch., FL 32014

McClellan, Jane
Ft. King Mid. Sch.
545 NE 17th Ave. Ocala, FL 32670

McFatter, William
Broward Co. Sch.
13320 SW 4th St.
Ft. Lauderdale, FL 33310

McHenry, Dr. Hugh
Memphis St. Univ.
Dept. of Math Sci.
Memphis, TN 38152

McWeeny, Mae
Sister Kenny Inst.
2727 Chicago Ave., S.
Minneapolis, MN 55407

Merrill, Dr. John A.
Ohio St. Univ.
Columbus, OH 43210

Monish, Dick
Pensacola Jr. College
1000 College Blvd.
Pensacola, FL 32504

Mosow, Dr. David K.
Roosevelt Sch. Dist.
6000 S. 7th St.
Phoenix, Arizona 85040

Muldrow, William R.
Fla. Dept. of Ed.
Knott Bldg./#255
Tallahassee, FL 32301

Muntner, Judie
850 Hungerford Dr.
Rockville, MD 20850

Muscat, Dr. Eugene J.
Univ. of San Francisco
San Francisco, CA 94117

Nuesse, Dr. David A.
Univ. of Wisconsin-Eau Claire
Dept. of Comp. Sci.
Eau Claire, WI 54701

Olkes, Alan T.
Dade Co. Schs.
13135 Coral Way
Miami, FL 33175

Pas, Dr. Eric I.
Duke Univ.
Dept. of Civil Eng.
Durham, NC 27706

Pennington, Billy
Mervyn Stone Lib.
University Station
Birmingham, AL 35294

Perry, Ted
San Juan United Sch. Dist.
6141 Sutter Ave.
Carmichael, CA 95608

Philipp, Charles
850 Hungerford Dr.
Rockville, MD 20850

Pinski, Ms. Mary
195 E. Schubert Ave.
Glendale Heights, IL 60137

Pollard, Dr. James P.
Nevada Dept. of Ed.
Capitol Complex
Carson City, Nevada 89710

Poole, Dr. Lawrence H.
N. Country Comm. Col.
Saranac Lake, NY 12983

Price, Drs. Samuel T. & Dillingham, Larry
Illinois St. Univ.
113 N. Adelaide
Normal, IL 61761

Raucher, Mr. Stephen M.
Montgomery Pub. Sch.
850 Hungerford Dr.
Rockville, MD 20853

Reid, George A.
Sarasota Co. Sch. Bo.
4747 S. Tamiami Trail
Sarasota, FL 33581

Richard, Dr. Tom
Bemidji St. Univ.
Hagg Sauer Hall
Bemidji, MN 56601

Roblyer, M.D.
ICON Enterprises
P.O. Box 13176
Tallahassee, FL 32308

Rodel, Lee
Hopkins Pub. Sch.
1001 HW 7
Hopkins, MN 55343

Rohm, Robert R.
Scott Foresman & Co.
1900 E. Lake Ave.
Glenview, IL 60025

Roser, Mr. Jeff
Valparaiso Elem.
379 Edge Ave.
Valparaiso, FL 32580

Rushinek, Drs. Sara & Avi
Univ. of Miami
P.O. Box 248237
Coral Gables, FL 33124

Saltz, Martin
WV Univ.
900 Dudley Ave.
Morgantown, WV 26505

Sarapata, Susan L.
John Hersey High
1900 E. Thomas Ave.
Arlington Heights, IL 60004

Schepis, Nick A.
Broward Co. School Bo.
1300 W. Las Olas Blvd.
Ft. Lauderdale, FL 33312

Schilmoeller, Kathryn J. & Gary L.
Univ. of Maine
Human Devel.
Orono, Maine 04469

Schwandt, Lynn C.
Price Lab. Sch.
Univ. of N. Iowa
Cedar Falls, Iowa 50613

Sexton, Porter
Portland Pub. Sch.
501 N. Dixon
Portland, OR 97227

Shostak, Robert
Fla. International Univ.
Tamiami Campus
Miami, FL 33199

Sloan, Sally A.
Minneapolis Pub. Sch.
807 NE Broadway
Minneapolis, MN 55413

Somers, Brenda S.
St. Louis Pub. Sch.
911 Locust, 4th Fl.
St. Louis, MO 63101

Spero, Dr. Samuel W.
Cuyahoga Comm. Col.
2900 Comm. Col. Ave.
Cleveland, OH 44115

Strong, Dr. Mary Winifred
GW Phillips Sch.
Rt. 1, Box 50
Morgantown, WV 26505

Stuart, Dr. Ann
Univ. of Evansville
P.O. Box 329
Evansville, IN 47702

Stutz, Dr. Joel
Univ. of Miami
P.O. Box 248237
Coral Gables, FL 33124

Swigger, Dr. Kathleen M.
N. Texas St. Univ.
NT Box 13886
Denton, TX 76203

Taber, Dr. Florence M.
Mall Plaza, Suite 250
157 S. Kalamazoo Mall
Kalamazoo, MI 49007

Taitt, Henry
Creative Programming
604 6th St.
Charleston, IL 61920

Thomas, Glenn
Fla. Dept. of Ed.
Pennington Bldg.
Tallahassee, FL 32301

Thomas, Rex
Iowa St. Univ.
Computation Ctr.
Ames, Iowa 50011

Tinsley, Jack T.
Univ. of N. Fla.
P.O. Box 17074
Jacksonville, FL 32216

Turner, Greg
Wayne Co. Sch. Dist.
33500 Van Born Rd.
Wayne, MI 48184

Umar, Amjad
Wayne Co. Mid. Sch.
P.O. Box 807
Wayne, MI 48184

Vinlove, Don
Valley View Sch. Dist 365U
636 Dalhart Ave.
Romeoville, IL 60441

Van Dusseldorp, Ralph
Univ. of Alaska
3221 Providence Dr.
Anchorage, Alaska 99454

Van Horn, Royal
Univ. of N. Fla.
Div. of Curri. & Inst.
Jacksonville, FL

Wall, Elizabeth S.
Bayshore Books
P.O. Box 848
Nokomis, FL 33555

Wallisch, Lt. Col., Wm. J.
USAF
Media Inst.
USAF Academy, Colorado 80840

Weaver, Catharine
McCorristin High
175 Leonard Ave.
Trenton, NJ 08610

Wetterlind, Dr. Peter
Univ. of N. Iowa
Dept. of Math
Cedar Falls, Iowa 50613

Wilde, Jane H.
Voorhees Col.
Curriculum Devel.
Denmark, SC 29042

Williams, Camelita
Reading Dept.
Norfolk St. Univ.
Norfolk, VA 23504

Willis, Cam
London Bo. of Ed.
165 Elmwood Ave.
London Ontario, Canada N6C 1J9

Winner, Alice-Ann
United Nations Intrnl. Sch.
24-50 E. River Dr.
New York, NY 10010

Wooley, William
Fla. Dept. of Ed.
Pennington Bldg.
Tallahassee, FL 32301

Wu, Dr. C.
U.S. Naval Academy
Mechanical Eng. Dept.
Annapolis, MD 21402

Young, Henderson
Davidson High
Mobile, Alabama

Zamecnik, Dr. Fred
N. Texas St. Univ.
NT Box 13886
Denton, TX 76203

Zuckerman, Dr. Robert A.
Dept. of Special Ed.
Kent St. Univ.
Kent, OH 44242